

Treball de Fi de Grau

Grau en Enginyeria en Tecnologies Industrials

Assistent d'organització de menús i aprofitament de recursos alimentaris (Appat).

MEMÒRIA

Autor: Jordi Granés Puig
Director: Lluís Solano Albajes
Convocatòria: Juny 2016



Escola Tècnica Superior
d'Enginyeria Industrial de Barcelona



Resum

El projecte Appat vol donar una solució local al problema global que és el malbaratament alimentari. Es defineix com a solució local pel fet que es focalitza només en una part del malbaratament alimentari, el que es produeix a nivell domèstic, el qual representa un 50% del total als països desenvolupats. En la major part dels casos aquest malbaratament és fruit de la caducitat no planificada dels aliments i la tasca d'Appat és ajudar als usuaris a organitzar millor els seus àpats per a poder consumir els aliments abans de que aquests es facin malbé.

Assistir a l'usuari perquè aquest pugui organitzar millor els seus àpats permet reduir el malbaratament de diverses maneres. En primer lloc, la possibilitat d'optimitzar els ingredients dels que es disposa fa que l'usuari necessiti menys estoc d'aliments, reduint conseqüentment la probabilitat de que aquests caduquin sense que ell se n'adoni. En segon lloc, l'aplicació conté una funció dissenyada específicament per a conèixer quins aliments permeten esgotar-ne un determinat, suposadament el que està a punt de caducar, a través d'un o múltiples àpats, on és l'ingredient principal. En tercer lloc, la possibilitat de poder introduir lliurement els ingredients dels que es disposa, permet a l'usuari buscar solucions per a l'aliment crític sense necessitat d'anar a comprar, sempre que tingui suficients ingredients a casa. Totes aquestes funcionalitats són aplicables més enllà de situacions on l'aliment caduca i poden resoldre combinatòries més efectivament del que ho faria una persona, especialment quan es tracta de múltiples àpats i múltiples comensals.

Aquesta solució s'ha posat en pràctica a través d'una aplicació d'ordinador, que posteriorment s'espera adaptar per a telèfons mòbils de sistema operatiu Android per a assolir major flexibilitat i nombre d'usuaris. Per a crear-la s'ha escollit el llenguatge de programació Python amb l'ajuda del seu mòdul Kivy per a la interfície d'usuari, treballat en un ordinador amb sistema operatiu Windows.

Sumari

| | |
|---|-----------|
| RESUM | 1 |
| SUMARI | 3 |
| 1. GLOSSARI | 7 |
| 2. PREFACI | 8 |
| 2.1. Origen del projecte | 8 |
| 2.2. Motivació | 8 |
| 2.3. Requeriments previs | 8 |
| 3. INTRODUCCIÓ | 9 |
| 3.1. Objectius del projecte | 9 |
| 3.1.1. Objectius acadèmics | 9 |
| 3.1.2. Objectius de l'aplicació..... | 9 |
| 3.2. Abast del projecte | 10 |
| 4. ESTAT DE L'ART | 11 |
| 4.1. Supercook..... | 11 |
| 4.2. Epicurious | 12 |
| 4.3. AllRecipes.com..... | 12 |
| 5. EINES DE PROGRAMACIÓ | 13 |
| 5.1. Python (2.7.11)..... | 13 |
| 5.2. Kivy (1.9.1) | 13 |
| 5.2.1. Aprenentatge del llenguatge Kivy | 13 |
| 5.2.2. Problemes amb el Kivy | 17 |
| 5.3. Lligam Python + Kivy..... | 18 |
| 5.4. Alternatives..... | 18 |
| 5.4.1. Java (app)..... | 19 |
| 5.4.2. HTML + JavaScript + CSS (webapp)..... | 19 |
| 5.4.3. PyQt/ PySide (programa d'ordinador)..... | 19 |
| 6. DISSENY PRELIMINAR | 20 |
| 6.1. Base de dades | 20 |
| 6.2. Funcionalitats | 21 |
| 6.3. Heurística..... | 21 |
| 6.4. Interfície d'usuari | 22 |

| | |
|---|-----------|
| 7. APPAT | 24 |
| 7.1. Funcionalitats | 24 |
| 7.1.1. Secció “Un Sol àpat” | 24 |
| 7.1.2. Secció “Múltiples àpats” | 26 |
| 7.1.3. Secció “Em vull acabar...” | 29 |
| 7.2. Heurística | 29 |
| 7.3. Interfície d’usuari | 30 |
| 7.4. Informació a l’usuari | 32 |
| 7.5. Altres | 33 |
| 7.5.1. Funcions de tractament d’errors/excepcions | 33 |
| 7.5.2. Implementació en telèfon mòbil | 34 |
| 7.5.3. Llengua | 34 |
| 7.5.4. Assumpcions i informacions diverses | 35 |
| 8. BASE DE DADES | 37 |
| 8.1. Implementació | 37 |
| 8.2. Contingut | 38 |
| 8.3. Base de dades en format diccionari | 39 |
| 8.3.1. Diccionari plats a ingredients | 39 |
| 8.3.2. Diccionari ingredients a plats | 40 |
| 9. PROVES | 41 |
| 9.1. Criteri | 41 |
| 9.2. Paràmetres de les proves | 42 |
| 9.2.1. Un sol àpat | 42 |
| 9.2.2. Múltiples àpats | 44 |
| 9.3. Execució i resultats de les proves | 46 |
| 9.3.1. Un sol àpat | 47 |
| 9.3.2. Múltiples àpats | 48 |
| 10. PLANIFICACIÓ TEMPORAL I COSTS | 50 |
| 10.1. Planificació temporal | 50 |
| 10.2. Costs | 50 |
| 10.2.1. Propietat intel·lectual | 50 |
| 10.2.2. Servidors i Google play | 51 |
| 10.2.3. Mà d’obra | 51 |
| 10.2.4. Costs totals | 52 |
| 11. IMPACTE MEDIAMBIENTAL | 53 |

| | |
|------------------------------------|-----------|
| 12. VERSIONS FUTURES | 54 |
| 12.1. Funcionalitats | 54 |
| 12.2. Heurística..... | 55 |
| 12.3. Interfície d'usuari | 57 |
| 12.4. Informació a l'usuari | 58 |
| 12.5. Altres | 58 |
| 13. CONCLUSIONS | 59 |
| 13.1. Compliment d'objectius | 59 |
| 13.2. Aprenentatge | 59 |
| 13.3. Observacions finals | 61 |
| AGRAÏMENTS | 63 |
| BIBLIOGRAFIA | 64 |
| Referències bibliogràfiques | 64 |

1. Glossari

- **App:** Aplicació.
- **ASCII:** American Standard Code for Information Interchange. És el codi que permet als ordinadors convertir els senyals binaris en lletres, números i símbols.
- **GETI:** Grau en Enginyeria en Tecnologies Industrials.
- **GUI (anglicisme):** Graphical User Interface o Interfície Gràfica d'Usuari.
- **NUI (anglicisme):** Natural User Interface o Interfície Natural d'Usuari
- **PIP:** Paquet de gestió de la instal·lació de paquets de software escrits en Python
- **TFG:** Treball de final de grau.
- **TUI (anglicisme):** Touch User Interface o Interfície Tàctil d'Usuari
- **Wheel (programació):** Format per a paquets de Python dissenyat per a contenir els fitxers d'instal·lació d'un programa o paquet.
- **Widget (anglicisme):** Un widget és un element d'una Interfície Gràfica d'Usuari (GUI) que mostra informació o bé permet a l'usuari interactuar amb un Sistema operatiu o una aplicació.
- **.apk:** Terminació dels paquets i aplicacions Android.
- **.kv:** Terminació dels fitxers Kivy.
- **.py:** Terminació dels fitxers Python.
- **.txt:** Terminació dels fitxers de text.

2. Prefaci

2.1. Origen del projecte

El projecte Appat neix com a idea espontània del seu creador, Jordi Granés Puig, davant d'una problemàtica real en la qual era necessari gestionar els menús dels següents dies per a dues persones. En el marc d'una estada de turisme, s'havia comprat massa menjar i era necessari gestionar els menús de la millor manera possible per tal que sobressin només aquells ingredients que podrien resistir el viatge de tornada a Barcelona. Així naixia la idea d'Appat.

2.2. Motivació

Citant el resum inicial, a grans trets "El projecte Appat vol donar una solució local al problema global que és el malbaratament alimentari". Es busca combatre la problemàtica del malbaratament alimentari ja que aquesta és un absolut despropòsit que no aporta res a la societat i, per contra, té un fort impacte en els cada cop més febles ecosistemes que ens envolten. S'espera doncs que, gràcies a una millor organització dels àpats, els aliments no assoleixin la seva data de caducitat abans de ser menjats, evitant així que s'hagi de llençar menjar.

2.3. Requeriments previs

Treballar en l'aplicació Appat, ha requerit l'aprenentatge de competències bàsiques de programació en els llenguatges Kivy i Python. Mentre que les competències bàsiques de programació en llenguatge Python ja estaven assolides gràcies a les assignatures del GETI "Fonaments d'informàtica" i "informàtica", l'aprenentatge del llenguatge Kivy s'ha fet de manera autònoma mitjançant el llibre "Creating Apps in Kivy" i la pàgina web de Kivy, [1] i [4]. Per a més informació, la secció 5.2 tracta sobre l'aprenentatge del llenguatge Kivy.

De la mateixa manera que succeeix amb Python, i altres programes de programari lliure, per a aprendre el llenguatge Kivy ha estat necessari tenir un nivell d'anglès suficient per a consultar les fonts d'informació, que gairebé sempre estan exclusivament disponibles en aquest idioma.

3. Introducció

3.1. Objectius del projecte

La gestió de projectes destaca la importància de definir correctament els objectius d'un projecte abans de començar a treballar-hi. Malgrat que sovint no és així, és important definir primer els objectius del projecte i després buscar maneres d'assolir-los per tal de poder proporcionar la millor solució possible a la problemàtica objectiu. No obstant, en tractar-se d'un treball de final de grau definit com a creació d'una App, els objectius fan referència a aquesta i no a la problemàtica real a la qual Appat dona solucions. Conseqüentment, els objectius d'aquest treball es poden dividir en dos grans blocs: D'una banda hi ha els objectius acadèmics, o objectius del tutor, i d'altra banda hi ha els objectius de l'aplicació, o objectius de l'alumne.

3.1.1. Objectius acadèmics

Pel que fa al tutor del treball, Lluís Solano Albajes, els seus objectius eren bàsicament observar i avaluar el funcionament del llenguatge de programació Python i la seva llibreria Kivy en un sistema operatiu Windows. Se sap que el Windows, especialment el Windows 10 per ser una versió recent, pot donar problemes amb tot el programari general que no estigui específicament dissenyat per a aquest sistema operatiu. És per això que s'han definit aquests programes per al projecte, perquè l'única manera de conèixer de primera mà quins contratemps es generen i quin és el seu impacte en el resultat final és el desenvolupament de l'aplicació en aquestes condicions.

3.1.2. Objectius de l'aplicació

Com bé s'ha comentat en el resum general, l'objectiu del projecte el fer una aplicació que ajudi a gestionar correctament el rebost de l'usuari, permetent així reduir a la mínima expressió el malbaratament causat per la caducitat dels aliments. S'espera que, més enllà de funcions secundàries que pugui contenir, sigui capaç de retornar correctament diferents combinacions de plats a partir d'una quantitat d'ingredients, un nombre de comensals i un nombre d'àpats aportats per l'usuari. Addicionalment es vol que l'aplicació estigui a l'abast de tothom pel que fa a competències culinàries i, és per això que els seus plats són senzills i requereixen pocs ingredients.

3.2. Abast del projecte

Degut a que s'ha treballat amb una metodologia àgil, l'abast del projecte és quelcom que s'ha anat definint a mesura que aquest es desenvolupava. S'entén com a metodologia àgil el procés de desenvolupament de programari que es basa en l'adaptabilitat davant qualsevol canvi com a mitjà per augmentar les possibilitats d'èxit del projecte. Habitualment les metodologies àgils es treballen per cicles iteratius que en aquest cas tenien una durada d'unes dues setmanes, temps que transcorria entre dues trobades alumne-tutor.

Pel que fa al desenvolupament en un primer moment es va definir com a funcionalitat bàsica la capacitat de suggerir, de manera efectiva, una combinació de plats per a múltiples àpats i múltiples comensals a partir d'una llista d'ingredients amb les seves quantitats i una base de dades. Un cop aquesta funcionalitat va estar desenvolupada, es van començar a introduir complements i funcionalitats addicionals que poden donar suport a aquesta funció principal o, simplement, sumar en la mateixa direcció d'evitar el malbaratament alimentari. D'aquesta manera, en acabar el projecte es disposa d'una aplicació capaç de:

- Suggerir de manera efectiva dues combinacions de plats per a múltiples àpats i múltiples comensals.
- Suggerir l'assignació dels ingredients sobrants com a acompanyaments dels diversos plats principals.
- Suggerir de manera ràpida i eficaç diversos plats per a un sol àpat i un sol comensal fent, a continuació, diverses assignacions d'acompanyaments sobre els tres plats que esgoten més ingredients.
- Mostrar plats que poden permetre a l'usuari esgotar un determinat ingredient.
- Mostrar receptes d'alguns dels plats continguts en l'aplicació.
- Donar informació útil a l'usuari sobre el funcionament general de l'aplicació.

S'espera poder ampliar l'abast del projecte en properes versions d'aquest, veure secció 12, però sobretot farà falta ampliar l'abast de mercat, és a dir el seu nombre d'usuaris potencial i real, per a poder combatre efectivament al malbaratament alimentari.

4. Estat de l'art

Abans de començar qualsevol projecte, és necessari fer una cerca exhaustiva per a garantir que el que es vol dur a terme no hagi estat desenvolupat amb anterioritat. En aquest cas, aquesta cerca va portar a la localització de diverses aplicacions amb característiques similars al que es volia que fos Appat, però cap d'elles tenia el mateix enfocament. Així com Appat és una aplicació que busca reduir el malbaratament alimentari a través de la organització dels menús dins d'una finestra de temps de múltiples àpats, la resta d'aplicacions del mercat simplement reben ingredients i retornen possibles combinacions de plats i menús per a un sol àpat. Queda palès que Appat és una aplicació clarament diferenciada de les alternatives del mercat però de totes maneres es procedirà a comentar les aplicacions presents al mercat per a que el lector pugui avaluar amb el seu propi criteri fins a quin punt són diferents les aplicacions.

Aquesta secció s'ha anat complementant a mida que es desenvolupava l'aplicació i per això s'hi veuran referències a seccions del projecte de les quals encara no s'ha parlat a la memòria. Pot ser interessant per al lector fer un cop d'ull a l'aplicació abans de començar a llegir aquesta secció.

4.1. Supercook

Supercook és una pàgina web d'accés gratuït que a partir d'una sèrie d'ingredients que l'usuari ha d'anar seleccionant d'entre uns de proposats, retorna una llista de possibles plats que es poden cuinar amb aquests ingredients. Aquests plats s'extreuen de diverses pàgines web sobre gastronomia, de manera que habitualment també es pot accedir a la recepta d'aquests clicant-hi a sobre. El fet de buscar els plats en pàgines web permet a Supercook tenir una base de dades colossal en comparació amb la d'Appat, de l'ordre de 15.000 plats comptant primers, segons i postres.

Pel que fa a la cerca de plats, Supercook ofereix dues funcionalitats molt interessants. En primer lloc, permet a l'usuari definir un ingredient que vol que surti a totes les receptes suggerides, convertint-la en una versió millorada de l'apartat "Em vull acabar..." d'Appat. En segon lloc, la funció de cerca de Supercook també retorna plats pels quals no es tenen tots els ingredients necessaris indicant quins són aquests ingredients que manquen. Aquesta funció és compatible amb la que permet a l'usuari definir un ingredient fix que vol esgotar i això li permet tenir una funcionalitat que engloba i perfecciona els apartats "Em vull acabar..." i "Un sol àpat".

Així doncs Supercook és una alternativa interessant a Appat pel que fa a les seves

funcions secundàries, “Em vull acabar” i “Un sol àpat” però no és capaç d’entrar en la planificació a llarg termini, funcionalitat principal d’Appat.

Nota: No confondre supercook.com amb supercook.me. La primera és la pàgina de la qual s’està parlant i la segona és una pàgina, que també té aplicació mòbil, de suport al robot de cuina Supercook.

4.2. Epicurious

Epicurious és una aplicació gratuïta només disponible per a dispositius Apple. Per culpa de no disposar d’un d’aquests dispositius, s’ha hagut de consultar tota la informació referent a ella per internet. Segons digitalTrends, [10], Epicurious permet definir un ingredient principal i a partir d’altres variables determinades per l’usuari li suggereix un àpat que pugui cuinar. Textualment s’afirma que Epicurious respon més aviat a la pregunta “Què puc fer amb X?” que “Què puc fer amb el que tinc”.

Epicurious també disposa d’una pàgina web plena de receptes i articles sobre cuina juntament amb un creador de menús. Malauradament aquest creador de menús es tracta d’una secció on l’usuari pot escriure plats i consultar-los, però en cap cas ajuda a la logística de menús a partir dels ingredients disponibles. Queda clar, doncs, que Appat i Epicurious són aplicacions que no persegueixen el mateix objectiu i públic.

4.3. AllRecipes.com

AllRecipes.com és una webapp, una app dins d’una pàgina web, que té unes funcionalitats semblants a les d’Epicurious. AllRecipes.com proporciona articles sobre cuina i receptes sent també capaç de suggerir plats que l’usuari pot cuinar a partir d’entre un i set ingredients principals. Aquesta pàgina també és una clara competidora de la funcionalitat “Un sol àpat” però, un cop més, no es focalitza en la gestió de múltiples àpats i múltiples comensals i per tant no comparteix ni públic ni objectius amb Appat. Per últim, és interessant prendre nota d’aquesta aplicació, ja que compta amb filtres predeterminats en funció d’una sèrie de restriccions alimentàries habituals com poden ser vegans, vegetarians, celíacs...

5. Eines de programació

Un cop decidit que la solució local al problema del malbaratament alimentari és la creació d'una aplicació, sorgeix la qüestió de quin software s'utilitza per a desenvolupar el projecte. Aquesta decisió ha estat presa per part del tutor ja que, tal i com es comenta a la secció 3.1.1, s'ha cregut convenient explorar el funcionament de Python com a llenguatge de programació dins d'un ordinador de sistema operatiu Windows i utilitzar la seva llibreria Kivy per a crear l'app. Per a la redacció i estructuració del codi, s'ha utilitzat l'editor de text Spyder, contingut en el conjunt Anaconda.

5.1. Python (2.7.11)

Python és un llenguatge de programació orientat a objectes gratuït i de font oberta. La seva filosofia de disseny emfatitza la llegibilitat del codi, i la seva sintaxi permet escriure programes més curts que els seus homòlegs en C++ o Java. Més enllà de les seves virtuts, també cal dir que la comunicació entre Python 2.7.11 i Windows 10 genera certs problemes amb els accents, i els caràcters en general de l'ASCII estès, que es comenten amb detall a la secció 8.1.

La instal·lació de Python en un ordinador de sistema operatiu Windows no és una tasca especialment senzilla. La instal·lació de Python és un procés molt personalitzable que fa que l'usuari hagi de prendre més decisions del que és habitual en la instal·lació d'un programa, i això fa que sigui un procés confús per als usuaris no experimentats. No obstant, la seva instal·lació és molt més senzilla que la del mòdul Kivy gràcies a l'existència d'installers.

5.2. Kivy (1.9.1)

Kivy és una llibreria de programari obert que serveix per a desenvolupar aplicacions per a mòbil o altres dispositius amb TUI (interfície tàctil d'usuari) a través d'una NUI (interfície d'usuari natural). S'ha escollit utilitzar la llibreria Kivy perquè era un dels objectius acadèmics del treball però realment és un mòdul molt interessant de treballar, especialment si es preveu que l'aplicació es pugui acabar implementant en un dispositiu tàctil, ja que està dissenyat per a facilitar la seva conversió al format necessari.

5.2.1. Aprenentatge del llenguatge Kivy

Com ja s'ha comentat a la secció 2.3, la creació de l'aplicació Appat ha requerit

l'aprenentatge de competències bàsiques en el llenguatge Kivy. Per evitar malentesos cal dir que Kivy no és pas un llenguatge de programació, sinó una llibreria complementària a Python que serveix per a poder crear interfícies d'usuari d'aplicacions d'una manera ràpida i senzilla. No obstant, Kivy té un funcionament una mica peculiar ja que, sempre amb la intenció de facilitar les coses al programador, treballa amb un llenguatge diferent que Python i en un fitxer diferent, tal i com es comenta a la secció 5.3. A continuació se'n pot veure un exemple explicat pas a pas:

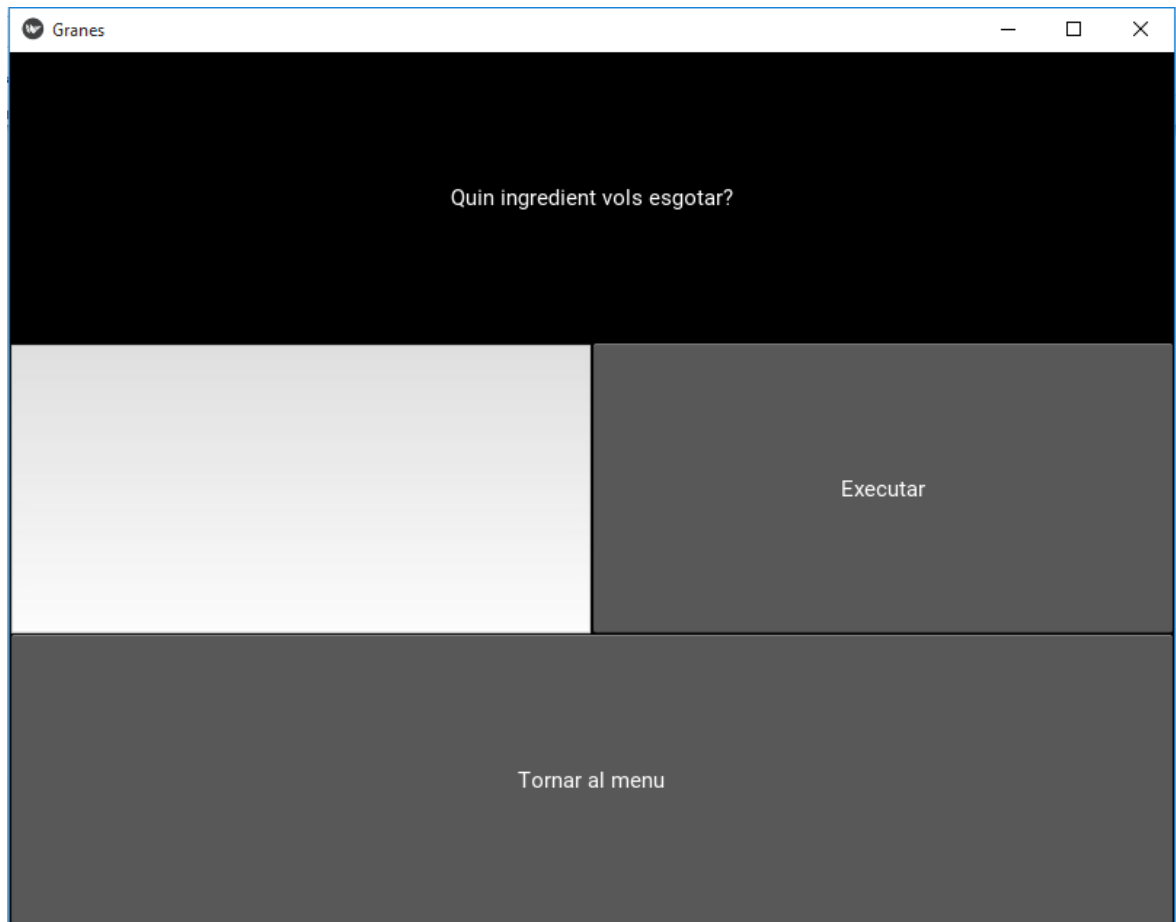
<Acabar>:

```

search_input: search_box
etiqueta:etiquetaVariant
BoxLayout:
    orientation: 'vertical'
    Label:
        id: etiquetaVariant
        text: 'Quin ingredient vols esgotar?'
    BoxLayout:
        TextInput:
            id: search_box
        Button:
            text:'Executar'
            on_press:
                if str(root.search_input.text).lower() in app.diccionariIngredientsPlats: app.llistaAcabar=
app.acabar1(app.diccionariIngredientsPlats,str(root.search_input.text).lower())
                ;
root.manager.current='Acabar2' ; root.manager.transition.direction='right' ; root.search_input.text=""
                else: root.etiqueta.text='Ho sentim, aquest ingredient no apareix a la base de dades'
    Button:
        text: 'Tornar al menu'
        on_press:
            root.search_input.text=""
            root.etiqueta.text= 'Quin ingredient vols esgotar?'
            root.manager.current = 'Menu'
            root.manager.transition.direction = 'left'

```

I aquesta part del codi genera la següent pantalla:



Com es pot veure, aquest exemple correspon a la pantalla de la funció “Em vull acabar...” i és una bona síntesi del que és el llenguatge Kivy.

Abans de començar, cal destacar que el llenguatge Kivy funciona amb classes que es creen al fitxer de llenguatge Python però es personalitzen al de llenguatge Kivy. La majoria d'aquestes classes, es defineixen per crear noves pantalles tot i que no és necessari definir una classe diferent per a cada pantalla, però sí molt recomanable des del punt de vista de l'autor. Aquestes classes creades per a cada pantalla heretaran de la classe Screen, definida pel llenguatge Kivy. També es crearan classes pels widgets que es vulguin personalitzar i una classe general per a l'aplicació, que heretarà de la classe App.

En primer lloc, en el fitxer .kv, el fitxer de llenguatge Kivy, cal definir el nom de la pantalla entre <> abans de començar a definir-ne els paràmetres com es pot veure en el codi d'exemple que es mostra, el qual comença per <Acabar>. Un cop definida la pantalla, s'hi poden començar a ubicar widgets, tot i que cal identificar-los dins de la classe si se'ls vol modificar com a conseqüència d'esdeveniments activats per l'usuari. Aquesta identificació

de les variables es fa donant una id: `nomGlobalVariable` dins el widget i assignant-li un nom a nivell local de la classe definint `"nomLocalVariable : nomGlobalVariable"`. És interessant destacar que aquest tipus d'assignacions en el llenguatge Kivy es fan amb : enlloc d'un =, tot i que també s'utilitza = per a altres funcionalitats.

Un cop identificats els widgets necessaris, la ubicació d'aquests és un dels majors problemes que l'autor s'ha trobat desenvolupant l'app, doncs succeeix que les comandes de posicionament dinàmic `pos_hint_x` i `pos_hint_y` no funcionen correctament. Aquestes comandes haurien d'ubicar els widgets al llarg de la pantalla amb un posicionament dinàmic que depengués de la mida de la pantalla i, per contra, quan s'intenten utilitzar queden tots els widgets apilats a la cantonada inferior esquerra. Se sospita que el culpable d'aquesta disfunció és la relació entre Python 2.7 i Windows i per evitar-la només hi ha dues alternatives.

La primera alternativa és fer un posicionament absolut de les variables, per píxels. Es pot veure que això no és intel·ligent en una app que en un futur es voldria implementar en un telèfon mòbil, i és per això que s'ha optat per la segona alternativa, ubicar tots els widgets amb un `BoxLayout`. El `BoxLayout` és una comanda del Kivy que divideix l'espai que se li dóna a parts iguals entre els widgets que ha de contenir. Aquest repartiment de l'espai pot ser horitzontal o vertical i aquesta capacitat, juntament amb la possibilitat de posar un `BoxLayout` dins d'un `BoxLayout`, ha permès organitzar la interfície satisfactòriament tot i els problemes prèviament esmentats. Com es pot veure en aquest exemple, la pantalla s'ha dividit en 3 espais pel que fa a la orientació vertical i l'espai central s'ha dividit en dos espais distribuïts horitzontalment.

Després de definir el `BoxLayout` i la seva "orientation" com a "vertical" o "horizontal", és moment de determinar quins i quants widgets ocuparan els espais. Pel desenvolupament de tota l'aplicació Appat s'han utilitzat quatre tipus de widgets. Els tres més comuns són els que es poden veure en aquest exemple: Els "Button" o botó; els "Label" o etiqueta i els "TextInput" o input de text. El quart widget és el "DropDown" o desplegable i permet que en clicar un botó determinat, s'activi un desplegable de botons a sota seu.

Un cop escollits els widgets cal definir les seves propietats o atributs que es necessitin, no s'ha d'oblidar que els widgets també són classes. S'ha optat només per definir els paràmetres indispensables per a cada widget, tot i saber que es poden definir més coses com ara el color d'aquest. En aquest exemple només s'ha definit l'atribut text dels tres tipus de widgets.

Finalment cal definir les accions a executar quan es faci clic sobre els botons. D'una banda el botó tornar al menú reinicia les variables d'aquesta pàgina i trasllada a la pantalla

del menú inicial. D'altra banda el botó executar trasllada a la següent pantalla, mentre executa funcions i emmagatzema variables necessàries per aquesta. És important fixar-se si l'ingredient determinat apareix o no entre els ingredients amb els que treballa l'app per evitar problemes amb les funcions que es criden en clicar-lo. Afortunadament, Appat està programada per donar una alerta a l'usuari si el nom de l'ingredient que s'ha introduït no correspon a cap dels de la base de dades i no continua amb el procés habitual si això passa. Aquesta informació es mostra a través de l'etiqueta situada a la part superior.

Per últim, hom s'haurà adonat de l'existència d'una paraula estranya a l'exemple: La paraula root. Root fa referència a la pantalla on es troba l'aplicació, de manera que és una paraula clau pel que fa a la crida de variables definides localment per a aquesta pantalla. Així doncs si es vol modificar el text de la variable local anomenada "etiqueta" caldrà escriure `root.etiqueta.text = "elNouTextQueEsDesitgi"` tot i que també es podria fer sobre la variable global escrivint `etiquetaVariant.text = "elNouTextQueEsDesitgi"`. Root també s'utilitza per part de l'screen manager, per redefinir la nova pantalla quan es vol fer un canvi amb la comanda `root.manager.current = "nomDeLaNovaPantalla"`.

5.2.2. Problemes amb el Kivy

Treballar amb Kivy no ha estat un camí de roses començant pel procés d'instal·lació. Així com existeix un assistent d'instal·lació per al Python en sistema operatiu Windows, el Kivy s'ha d'instal·lar donant indicacions a través de l'indicador d'ordres de Windows i requereix que prèviament s'hagi instal·lat l'última versió de PIP i wheel. Aquest procés és quelcom totalment diferent als habituals processos d'instal·lació a través d'assistents i si un no hi està acostumat, no resulta fàcil. Afortunadament el procés està explicat amb suficient detall a la pàgina web de Kivy, [4], com per acabar sortint-se'n.

Un cop superats els problemes en la instal·lació, també s'ha complicat l'aprenentatge del llenguatge Kivy. Lamentablement, la informació de les dues fons "oficials", el llibre "Creating Apps in Kivy" i la pàgina web de Kivy, [1] i [4], no és de gran qualitat. Això ha provocat que sovint hagi estat necessari consultar altres vies d'informació, principalment pàgines web sobre dubtes de programació com ara Stackoverflow, [5], i vídeos explicatius com els de l'usuari Sentdex, Harrison Kinsley, de Youtube. La baixa qualitat de les dues suposades fonts principals té diferents orígens per a cada una d'aquestes.

Pel que fa al llibre, el seu problema és que està totalment focalitzat en els nombrosos exemples que conté, els quals són molt interessants i enriquidors, però no dóna premisses generals per anar més enllà. Un clar exemple seria la lògica: El llibre conté una secció a la pàgina 22 anomenada "Adding Logic to a Custom Widget", on presumptament

s'hauria de parlar sobre incloure funcions lògiques en el fitxer .kv però realment l'únic que fa és recomanar aplicar la lògica en el fitxer Python i tractar un exemple sobre com fer una funció normal enlloc de dinàmica, sense explicar com es fa per incloure lògica dins del fitxer .kv. Un detall il·lustratiu és que a dia d'avui, l'autor encara no ha aconseguit esbrinar com s'introdueixen funcions lògiques dins d'una altra funció lògica en un fitxer Kivy.

Pel que fa a la pàgina web de Kivy, aquesta té dos inconvenients que n'impedeixen una eficiència real a l'hora d'informar a l'usuari. Primer de tot, es troba que les explicacions i els exemples dels atributs de les classes sovint són molt breus, cosa que fa que en alguns casos calgui buscar informació i exemples sobre aquests widgets en altres fonts. En segon lloc es troba el problema del llenguatge. Com ja s'ha comentat anteriorment, Kivy treballa amb un llenguatge de programació propi, però com que des del fitxer Python es poden editar les classes i definir noves funcions, sovint hi ha tres maneres diferents de fer les coses: Una des del fitxer Kivy amb llenguatge Kivy, una altra des del fitxer Python amb el llenguatge Python i una solució mixta. Això no és un problema en sí, ans al contrari, però el que sí que és un contratemps és que els exemples i explicacions de la web sovint estan només en un dels dos llenguatges de programació, de manera que si es tenia previst treballar la solució des de l'altre llenguatge aquesta informació no és útil.

5.3. Lligam Python + Kivy

Se suposa que quan hom treballa amb la dobleta Python+Kivy, ho ha de fer amb dos fitxers de text, un .py que contingui tot el codi Python i un .kv que contingui tot el codi Kivy amb una sèrie de normes perquè l'app sàpiga quins són els dos fitxers amb els que ha de treballar i pugui anar alternant les crides a tots dos. No obstant, l'autor va decidir unificar els dos documents en un sol fitxer .py a través de la comanda "Builder.load_string" que permet tenir el codi Kivy a dins del fitxer .py. Aquesta decisió es va prendre bàsicament per comoditat ja que no es volia treballar en dos documents diferents. És cert que això té una contrapartida i és que l'autor ha de ser ordenat a l'hora d'estructurar el codi, però si s'és capaç de mantenir un bon ordre, no hi ha cap altra de contrapartida aparent.

5.4. Alternatives

Hi ha certes combinacions alternatives de programari que s'haurien pogut utilitzar per a desenvolupar l'App enlloc de la dobleta Python+Kivy i en aquesta secció es comentaran breument algunes d'aquestes. Com es pot veure no totes les alternatives creen una app, però tant una app, com una webapp, com un programa d'ordinador haurien permès desenvolupar el projecte Appat.

5.4.1. Java (app)

Java és un llenguatge de programació concurrent, basat en classes i orientat a objectes. Està especialment dissenyat per a requerir el mínim de dependències d'implementació possibles i segueix la filosofia WORA (Writ Once, Run Anywhere). Això significa que un codi compilat en Java pot ser utilitzat en qualsevol plataforma que tingui el Java instal·lat sense necessitat d'una recompliació. Aquestes característiques converteixen a Java en un llenguatge de programació especialment adequat per a desenvolupar aplicacions de mòbil.

5.4.2. HTML + JavaScript + CSS (webapp)

Aquesta combinació treballa amb tres llenguatges que per sí sols no podrien crear una app, però que gràcies a la seva interacció en són capaços. Es tracta de crear una pàgina web amb el llenguatge HTML (HyperText Markup Language) que contingui scripts en Javascript per afectar al comportament de la web davant dels inputs de l'usuari, convertint el conjunt en una webapp. Per últim, s'utilitzaria el CSS (Cascading Styles Sheets) per a millorar la presentació de la interfície d'usuari, doncs HTML no està dissenyat per això i els seus propis creadors encoratgen a utilitzar CSS.

5.4.3. PyQt/ PySide (programa d'ordinador)

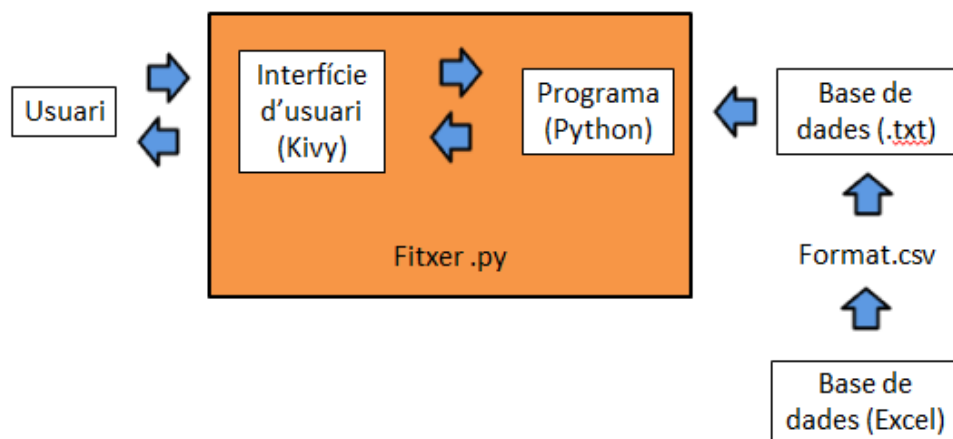
PyQt és un binding de la biblioteca gràfica Qt per a Python. En programació, un binding és quelcom que permet connectar dos llenguatges diferents, en aquest cas PyQt està connectant la biblioteca Qt amb el llenguatge de programació Python. Qt és una biblioteca que permet, bàsicament, generar la interfície d'usuari i gestionar el comportament davant els inputs de l'usuari en una aplicació.

En ser PyQt un binding de pagament, aquesta alternativa hauria requerit utilitzar PySide. PySide és, un binding similar a PyQt que l'empresa Nokia, propietària de Qt, va desenvolupar després d'intentar convèncer infructuosament a l'empresa propietària de PyQt perquè alliberés el seu programa per a usos no professionals sota llicència LGPL. Una llicència LGPL (Lesser General Public License) és una llicència lliure que permet que programes propietaris usin les llibreries i aplicacions sota aquesta llicència.

6. Disseny Preliminar

Abans de començar a treballar en l'aplicació Appat era necessari traçar unes línies mestres per treballar sobre uns bons fonaments. Aquest procés és extremadament important perquè, en cas de fer-se correctament, estalviarà moltes hores de treball en un futur. El problema per a l'aplicació Appat és que aquest procés requereix, bàsicament, d'experiència en el camp i la realitat en el moment de desenvolupar-se era exactament el contrari. No obstant, era indispensable definir certes idees sobre les quals començar a desenvolupar Appat i, partint de l'esquema de funcionament que s'havia decidit i que es presenta a continuació, es van pensar idees per a cada una de les seccions.

Esquema de funcionament Appat



6.1. Base de dades

La base de dades es tracta en més profunditat a la secció 8, però a grans trets és l'estructura de dades que permet al programa conèixer les relacions entre plats i ingredients. Aquest punt s'havia considerat des d'un primer moment i va ser la seva senzillesa la que va convèncer a l'autor a desenvolupar Appat. La base de dades de l'aplicació consisteix en dos simples diccionaris: Un primer que dels plats porti a llistes d'ingredients i un segon que dels ingredients porti a llistes dels plats que els contenen com

a ingredient principal. Aquests dos diccionaris combinats són els que permeten desenvolupar satisfactòriament tots els processos de cerca de menús i per a un funcionament més òptim de l'aplicació contenen receptes senzilles que requereixen d'ingredients comuns a les llars catalanes. D'ara en endavant es parlarà d'aquests diccionaris com a diccionari de plats a ingredients i diccionari d'ingredients a plats.

Més enllà de relacionar plats i ingredients, la base de dades conté una sèrie d'informacions addicionals necessàries per al desenvolupament de certes funcions. Aquestes informacions addicionals han estat presents des d'un primer moment i són: Els codis per indicar els ingredients optatius o acompanyaments; La quantitat de cada ingredient principal necessària; i els plats que actuen com a plats-acompanyaments.

En començar el projecte es va considerar la possibilitat de permetre a l'usuari afegir plats a la base de dades, no obstant, això s'ha desenvolupat i quedarà per a una versió futura.

6.2. Funcionalitats

Tal i com s'ha comentat abans, la prioritat del projecte Appat ha estat, des d'un primer moment, una secció que permetés organitzar múltiples àpats per a múltiples comensals. No obstant, quan es va començar a desenvolupar l'aplicació també s'havia decidit que s'inclouria una secció on, donat un aliment, es retornarien els plats que el contenien per poder esgotar aquest aliment encara que fos necessari anar a comprar més ingredients.

6.3. Heurística

En el moment de començar a desenvolupar l'aplicació no es tenia completament definida l'heurística que s'utilitzaria, però sí que s'havia decidit que la cerca es basaria en recórrer els plats que es poguessin fer amb l'ingredient més abundant com a ingredient principal i mirar si es tenien la resta d'ingredients necessaris. Basar l'heurística en l'ingredient més abundant va ser una idea intuïtiva que permetia donar algun tipus de prioritat a l'hora de recórrer els plats de cada un dels ingredients i evitar així una cerca basada en l'ordre d'introducció dels ingredients, quelcom poc rigorós.

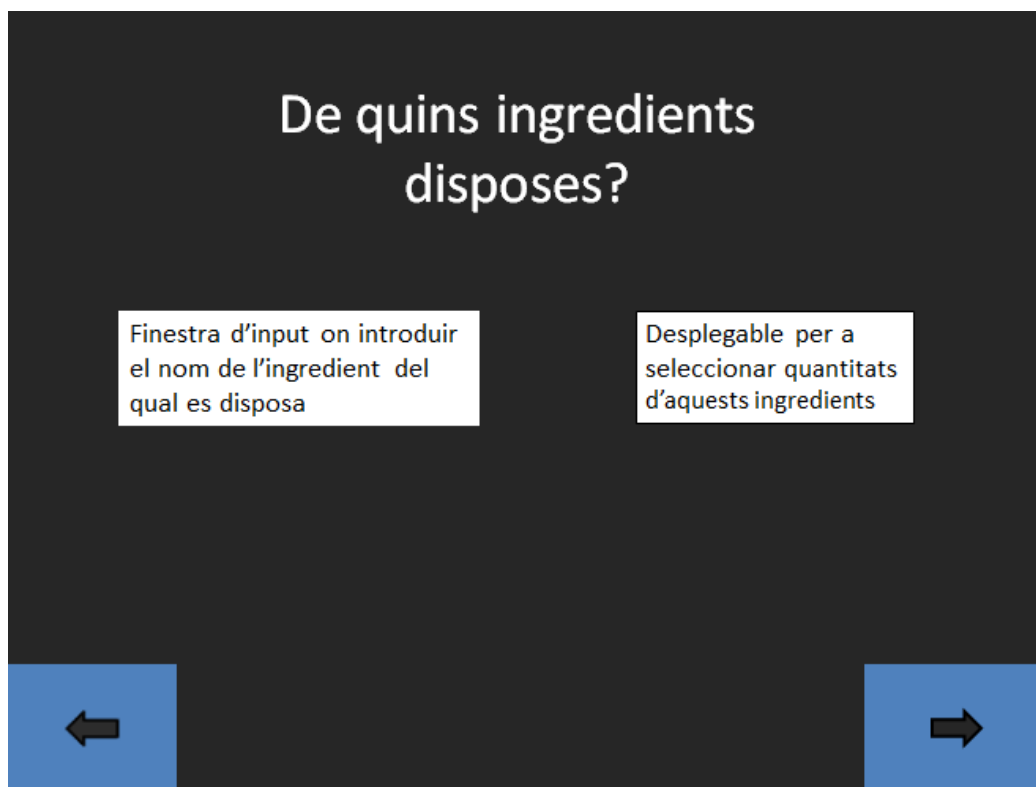
Pel que fa a l'assignació d'acompanyaments, des d'un principi s'ha tingut clar que aquests no es gestionarien de la mateixa manera que els ingredients principals, però no ha estat fins més endavant que s'ha definit la manera d'assignar-los.

6.4. Interfície d'usuari

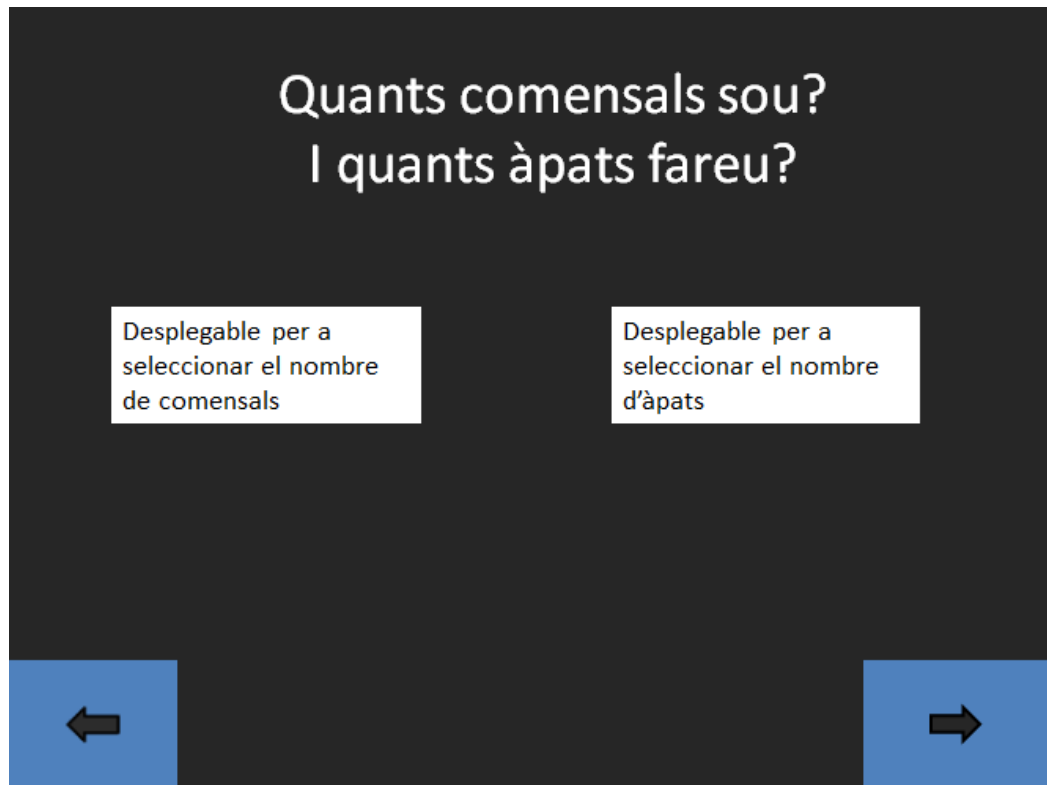
Com que abans de desenvolupar aquest projecte el tutor i l'alumne havien començat a desenvolupar una app sobre dades estadístiques, es va reciclar la interfície d'usuari primitiva que es tenia pel que fa a la pantalla de menús i que ha acabat sent la mateixa per a la versió final. Pel que fa a les seccions, no va ser fins que es van haver acabat les primeres versions de les tres funcionalitats bàsiques que se'n van fer els croquis sobre les seves interfícies.

A continuació es veu una representació a ordinador dels croquis que es van fer per a la secció "Múltiples àpats".

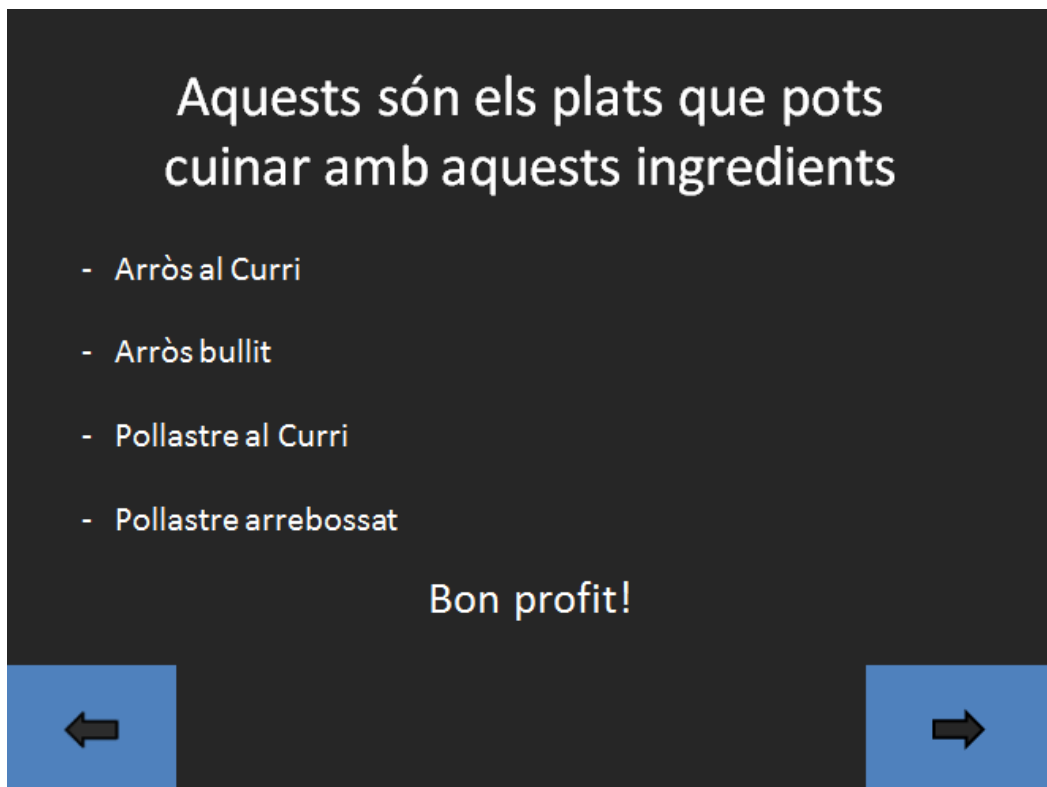
Pantalla 1: Introducció d'ingredients i quantitats. En seleccionar la fletxa cap a l'esquerra l'aplicació tornaria al menú i en seleccionar la fletxa cap a la dreta l'aplicació aniria a la pantalla 2.



Pantalla 2: Introducció del nombre de comensals i àpats. En seleccionar la fletxa a la dreta l'aplicació reconduiria a l'última pantalla de la secció que es pot veure a continuació. La fletxa a l'esquerra tornaria a la pantalla anterior.



Pantalla 3: Visualització dels resultats. En seleccionar la fletxa a la dreta es retornaria al menú principal, en seleccionar la fletxa a l'esquerra es tornaria a la pantalla anterior.



7. Appat

L'aplicació Appat es pot explicar des de les seves quatre branques principals: Les funcionalitats, l'heurística, la interfície d'usuari i la informació a l'usuari. A continuació es detalla l'estat de tots aquests punts i de les 11 funcions principals, d'un total de 29 que conté aquesta primera versió de l'aplicació.

7.1. Funcionalitats

Les tres seccions principals d'Appat depenen d'una sèrie de funcions, comunes entre elles o específiques de cada una, que permeten el seu funcionament i que són d'importància cabdal per a l'app.

7.1.1. Secció “Un Sol àpat”

Versió reduïda de la secció “Múltiples àpats” que permet a l'usuari conèixer quins plats pot cuinar amb els ingredients que té per casa. Pel que fa a l'assignació d'acompanyaments, l'aplicació selecciona els tres plats que, un cop assignats els acompanyaments, utilitzarien més ingredients i els mostra per ordre juntament amb els acompanyaments que se'ls poden assignar. Es busquen els plats que més ingredients esgoten per congeniar amb la idea de l'aplicació Appat, que busca consumir el major nombre d'aliments perquè aquests no es facin malbé. A la secció 13.3 es parla amb més profunditat sobre aquest tema, ja que consumir el major nombre d'aliments no és aparentment una solució al malbaratament alimentari.

Al contrari del que es pot esperar, aquesta secció no utilitza els mateixos algoritmes que la secció “Múltiples àpats” definint automàticament el nombre de comensals i àpats a u. Això és així ja que aquesta secció es va crear per experimentar i aprendre de cara a crear la secció “Múltiples àpats”. Així doncs s'esmentaran breument els algoritmes que utilitza aquesta secció, juntament amb els inputs de l'usuari i els dos diccionaris de la base de dades. Comentar que els noms de les funcions s'han citat textualment del codi per a no generar confusions si algú vol consultar-lo, però se sap que alguns no són noms del tot adients:

- **Funció actualitzaLlistaIngredients:** Donada una llista d'ingredients i un plat, actualitza la llista d'ingredients eliminant els ingredients necessaris per a aquest.
- **Funció assgnaAcompanyaments:** Funció que donada una llista de plats i una llista d'ingredients sobrants, retorna com es poden assignar els diversos

ingredients sobrants com a acompanyaments dels plats, en format llista. Com que és una funció que només suggereix l'assignació dels acompanyaments per deixar-la a criteri del/de la cuiner/a, aquesta actualitza la llista d'ingredients restant-hi els ingredients que s'utilitzen com a acompanyament. La funció fins i tot suggereix un acompanyament quan la quantitat restant d'aquest és zero. Això s'ha definit així ja que les sobres simbòliques d'un ingredient poden utilitzar-se com a acompanyament, doncs se sap que la quantitat d'un ingredient afegida com a acompanyament és molt variable. Per exemple un sol tomàquet cherry pot decorar una amanida o una gran quantitat de ceba sofregida pot tenir molta importància en un entrepà de Frankfurt.

- **Funció amanidesAndCo:** Busca quins plats-acompanyament es poden fer amb els ingredients dels que es disposa i busca si algun dels plats que es faran té algun d'aquests plats-acompanyament assignable. Assigna, si n'hi ha, els plats-acompanyament com a acompanyament dels plats, actualitza la llista d'ingredients i crida a la funció assignaAcompanyaments si encara sobren ingredients.
- **Funció build:** És la funció principal de l'aplicació i s'executa quan aquesta s'inicialitza. Com es pot intuir, s'anomena build ja que és la funció que s'executa en inicialitzar la classe mestra, GranesApp, la qual hereta de la classe App i crea l'aplicació. La seva funció és, més enllà d'inicialitzar la gestió de pantalles i l'aplicació, definir les variables globals i definir la finestra desplegable, o dropdown, per a poder utilitzar-la allà on sigui.
- **Funció cercaMatxaca:** Duu a terme la cerca dels plats que es poden fer amb els ingredients dels que es disposa. Bàsicament recorre per a cada un dels ingredients donats els plats que els contenen, a través del diccionari d'ingredients a plats, i s'ajuda de la funció pucFer per saber si es disposa dels ingredients necessaris per a fer aquest plat.
- **Funció emmagatzematge:** Emmagatzema el nom de l'ingredient que ha escrit l'usuari, si aquest està a la base de dades, en una llista d'ingredients que és variable global.
- **Funció ordena3PerOptim:** De tots els plats que es poden fer amb la combinació d'ingredients donada, determina els tres primers en funció de quants ingredients dels disponibles utilitzen. Si no es poden fer tres plats, conté funcions per evitar errors en l'app.
- **Funció pucFer:** A partir del nom d'un plat, la llista d'ingredients dels que es disposa amb les respectives quantitats i el nombre de gent que menjarà aquest

plat, retorna True si es disposa dels ingredients principals suficients per a cuinar-lo. En cas contrari retorna False.

7.1.2. Secció “Múltiples àpats”

Tal i com s’ha comentat prèviament, aquesta secció és el punt central del projecte Appat ja que és la que la diferencia d’altres aplicacions existents al mercat i li permet fer una aportació significativa a la lluita contra el malbaratament alimentari.

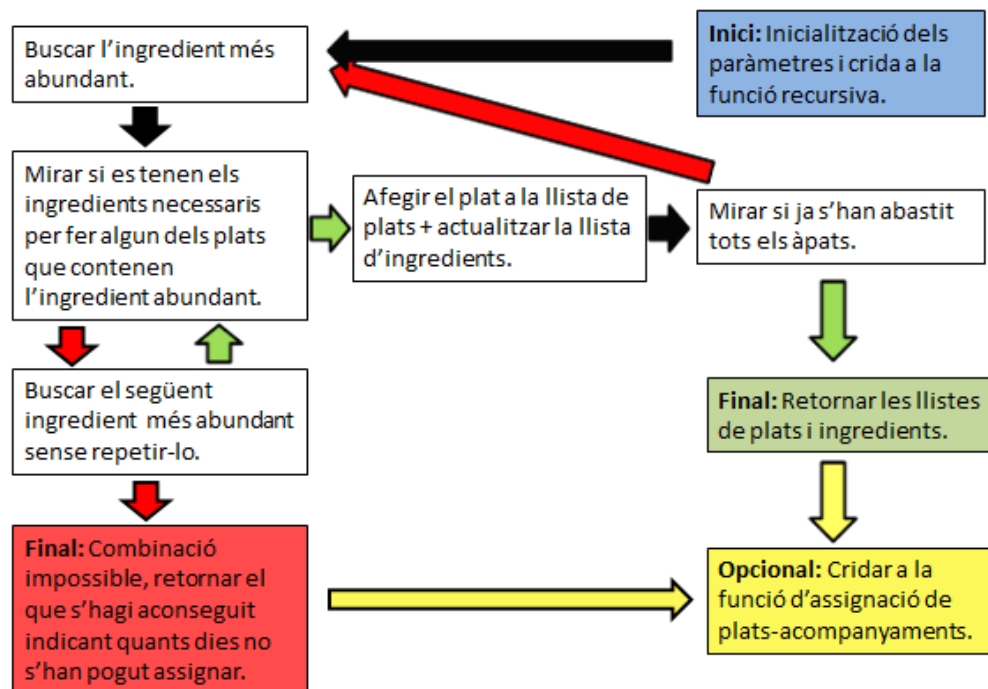
La secció es dedica a suggerir dues combinacions de plats que poden fer els usuaris, per a un nombre d’àpats i comensals, donats els ingredients dels que es disposen i la seva quantitat. Un cop definits els plats i els seus ingredients principals, una funció secundària es dedica a assignar els ingredients sobrants a cada un dels plats sense posar atenció a la quantitat per motius comentats a l’apartat anterior a la funció assignaAcompanyaments.

És important aturar-se a reflexionar sobre la gestió de les quantitats, doncs Appat utilitza un sistema innovador de racions. El sistema de racions va ser una invenció de l’autor per evitar introduir quantitats en forma de grams, com es faria en un programa tradicional. En comptes d’això l’usuari ha d’avaluar quantes racions té d’un ingredient determinat, entenent per ració la quantitat d’un ingredient que l’usuari considera que l’omple si pren un plat únic basat en aquest ingredient i amb algun acompanyaments. A continuació s’exposen els punts a favor d’aquest sistema que té com a única contrapartida que requereix a l’usuari tenir un cert criteri per avaluar les quantitats de menjar:

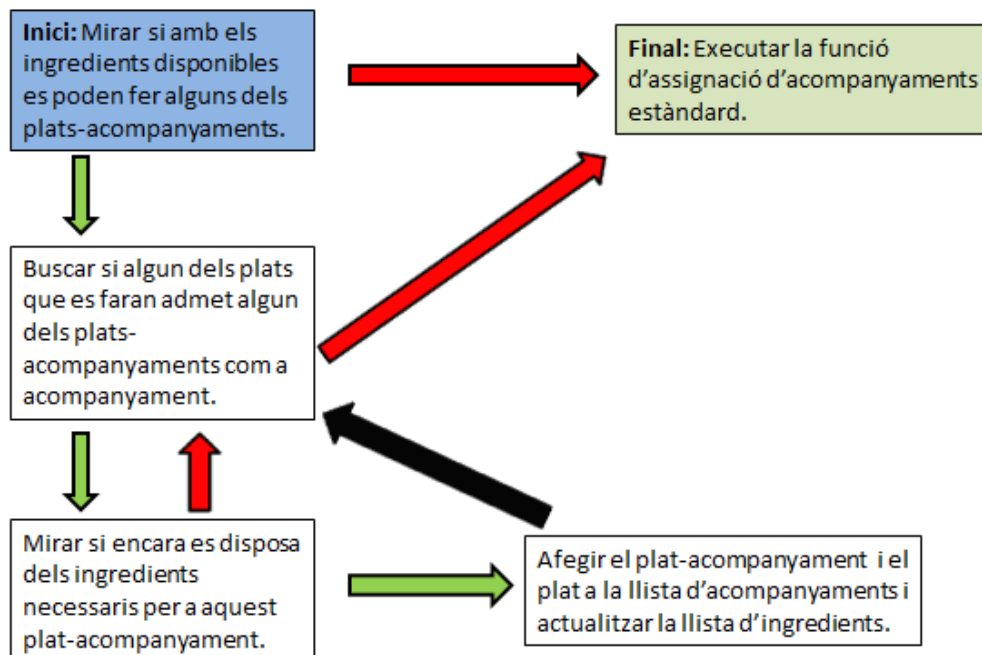
- Punts en contra del sistema de pes:
 - L’usuari necessita una balança per poder pesar els ingredients i perd temps per fer-ho.
 - Si la quantitat que aporta l’usuari no arriba al mínim estipulat per l’aplicació, per molt que faltin uns pocs grams aquesta considerarà que no és possible fer un plat determinat.
- Punts a favor del sistema de racions:
 - Les quantitats s’adapten a la gana que tingui cadascú. Hi ha persones per les quals un plat de 90 grams de pasta és suficient per a un àpat i altres que en necessiten 130.
 - Fa que l’usuari sigui responsable de la quantitat, convertint-lo també en responsable si li sobra o li falta menjar.

Abans d'explicar el funcionament exacte de la secció, és important destacar que l'heurística d'Appat considera que els comensals menjaran un plat únic. Si bé això seria modificable en una versió posterior, no és gens interessant pel que fa a assolir òptims. Si l'aplicació treballés amb àpats de dos plats es necessitarien més ingredients diferents per intentar evitar repetir plats en la combinatòria o, en cas contrari, l'usuari es trobaria amb l'aplicació suggerint-li repetir els plats. A partir d'aquesta assumpció, l'heurística escollida per a la resolució del projecte Appat és absolutament intuïtiva i es pot veure en el següent diagrama de flux.

Cerca múltiples àpats i comensals



Afegit de plats-acompanyaments



Per a poder desenvolupar aquest procés han estat necessàries les següents funcions principals. No s'explicaran les funcions més bàsiques i cal recordar que totes aquestes funcions tenen accés als dos diccionaris de la base de dades, el diccionari de plats a ingredients i el diccionari d'ingredients a plats:

- **Funció actualitzaLlistaIngredientsMultiple:** És la versió per a múltiples àpats de la funció actualitzaLlistaIngredients comentada a la secció anterior. No es pot utilitzar la mateixa versió, la múltiple, per les dues seccions ja que la secció "UnSol" no té en compte la quantitat dels ingredients.
- **Funció amanidesAndCo:** Veure la secció anterior
- **Funció assgnaAcompanyaments:** Veure la secció anterior
- **Funció build:** Veure la secció anterior
- **Funció cercaMultiplesComensals:** Inicialitza les variables i crida a la funció recursiva recursMultiplesComensals.
- **Funció emmagatzemaNomiQuant:** Versió de la funció emmagatzema de la secció anterior que permet emmagatzemar en una llista tant el nom d'un ingredient com la quantitat de la que es disposa.

- **Funció ingredientMesAbundant:** Donada una llista d'ingredients i la llista dels ingredients que ja han estat ingredients abundants, retorna un nou ingredient més abundant. En cas de no haver-hi un possible ingredient abundant no retorna res (None).
- **Funció pucFer:** Veure la secció anterior.
- **Funció recursMultiplesComensalsMultiplesApats:** Amb l'ajuda de les funcions descrites a continuació, gestiona tot el procés de cerca de plats i retorna la llista de plats, la quantitat d'àpats que queden per satisfer i la llista d'ingredients sobrants.

7.1.3. Secció “Em vull acabar...”

Aquesta és la secció més bàsica de l'aplicació i, bàsicament ensenya els plats que contenen un ingredient determinat per l'usuari. Serveix per a situacions en les quals l'usuari hagi d'esgotar un ingredient determinat encara que hagi d'anar a comprar altres aliments i això passa bàsicament quan l'usuari s'adona que un ingredient està a punt de caducar. Aquesta secció només utilitza una funció anomenada acabar1 que únicament retorna la llista corresponent a l'ingredient determinat al diccionari d'ingredients a plats.

7.2. Heurística

La resolució de la problemàtica d'assignar plats a cada un dels àpats, és un procés que no es regeix per la matemàtica lineal, ja que no és possible cap càlcul que ens permeti assolir una o més solucions igual de satisfactòries. És per això que es requereix de milers d'iteracions per tal de trobar alguna solució satisfactòria.

La iteració és necessària quan el problema a resoldre no és lineal i no és viable explorar totes les combinacions possibles. Com el lector haurà pogut apreciar, aquest és el cas d'Appat ja que el nombre total de combinacions es pot disparar molt ràpidament. A tall d'anècdota, és possible calcular el nombre d'iteracions que serien necessàries si es volgués explorar totes les combinacions possibles d'ingredients i comparar-les amb els plats de la base de dades. Per fer-ho cal aplicar el binomi de Newton assumint que s'utilitzen els 67 ingredients de la base de dades en combinacions de 2, 3 o 4 ingredients i es comparen amb els plats de la base de dades. A continuació se'n veuen els resultats:

67 ingredients possibles en combinacions de 2 = 2211 combinacions

67 ingredients possibles en combinacions de 3 = 47905 combinacions

67 ingredients possibles en combinacions de 4 = 766480 combinacions

Aquest nombre de combinacions pot semblar acceptable per a un ordinador modern, però cal recordar que aquest sistema seria incapaç de gestionar acompanyaments opcionals. A més, cal dir que la base de dades és experimental de manera que en una hipotètica implementació real del projecte Appat aquesta seria molt més gran, implicant una major quantitat de combinacions.

Un cop es fa evident la necessitat d'una heurística, cal dir que hi ha milers d'heurístiques diferents capaces de resoldre satisfactòriament el procés d'organització que duu a terme el projecte Appat. A grans trets, l'elecció de l'heurística ve determinada per dos factors: El temps d'execució i la precisió del resultat. A priori com més temps s'inverteixi en cercar solucions, més precisa serà aquesta, però per contra l'usuari vol gastar el mínim de temps executant l'aplicació. Afortunadament, una bona heurística permet reduir el temps de resolució alhora que s'incrementa la precisió del resultat i això es fa donant un criteri de cerca que permet descartar iteracions que no aporten res al procés.

Tal i com es preveia abans de materialitzar l'aplicació a la secció 6.3, l'heurística escollida es basa en l'ingredient més abundant. El programa sempre cerca els plats que contenen l'ingredient més abundant com a ingredient principal i mira si disposa dels ingredients per fer-ne algun. Si els té emmagatzema el plat, actualitza la llista d'ingredients i repeteix el procés fins que s'hagin assignat tots els àpats. Si no els té, repeteix la operació amb el següent ingredient més abundant.

Les proves de funcionament, veure secció 9, han demostrat que aquesta heurística és viable i suficientment efectiva. No obstant, aquesta sempre es podria millorar de cara a una versió futura.

7.3. Interfície d'usuari

Appat presenta una interfície d'usuari bastant senzilla però que permet moure's per les diverses pantalles i utilitzar les diverses funcionalitats sense problemes. Aspectes a destacar de la interfície d'usuari són:

- **Utilització de widgets personalitzats a les necessitats.** Per exemple s'ha creat la classe LabelRecepta per a definir etiquetes sistemàticament centrades, on el text no excedeixi l'espai assignat a l'etiqueta i on la mida del text estigui fixa a 15 punts.
- **Utilització d'una finestra desplegable (dropdown).** S'ha creat una finestra desplegable a la secció receptes i s'ha personalitzat per encaixar amb les exigències del projecte.

1- Netejar les patates, pelar-les i tallar-les a daus


2- Coure les patates a foc mig, tapades i amb oli abundant

3- Treure una bona part de l'oli, baixar el foc i afegir l'ou batut prèviament

4- Donar la volta a la truita amb l'ajuda d'un plat gran. Protegir-se les mans si és possible, doncs l'excident d'oli podria cremar-vos

5- Coure l'altre banda de la truita i deixar-la reposar 10 minuts abans de menjar

6- Bon profit!



Arròs al curri

Llenties

Pèsols amb bacó

Pollastre arrebossat

Pissots de bolets

- **Estructura simple i intuïtiva** que permet a l'usuari moure's còmodament pels menús. Es pot veure el menú principal com a exemple a continuació:

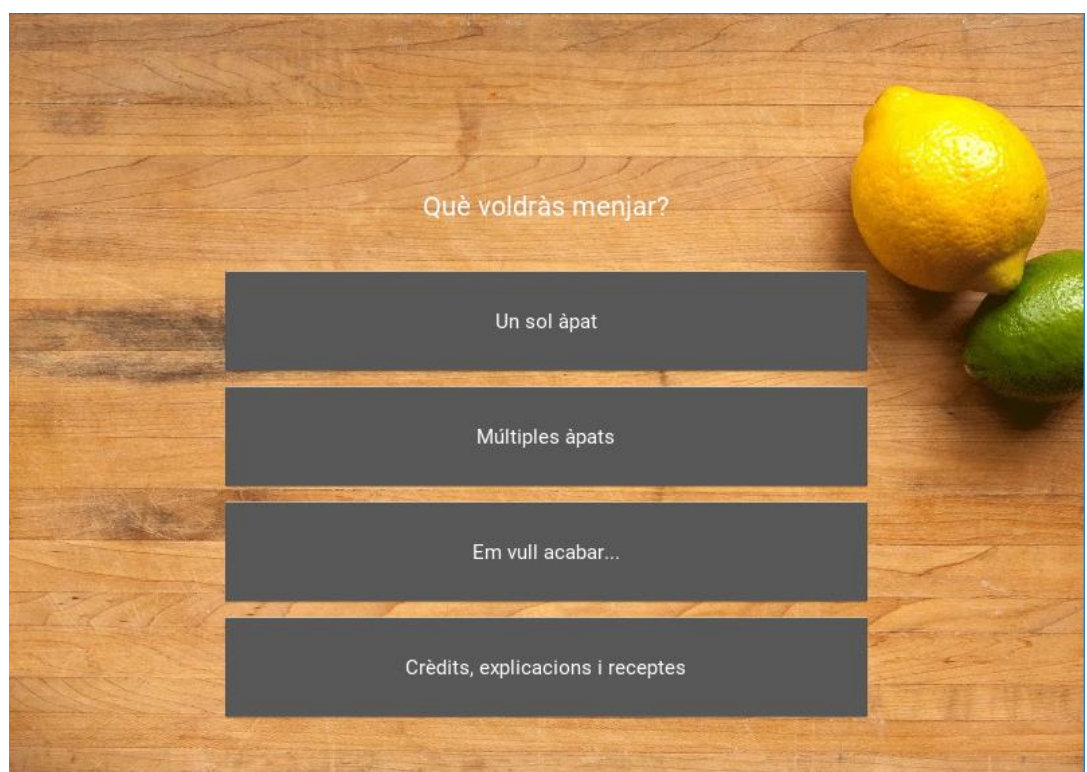
Què voldràs menjar?

Un sol àpat

Múltiples àpats

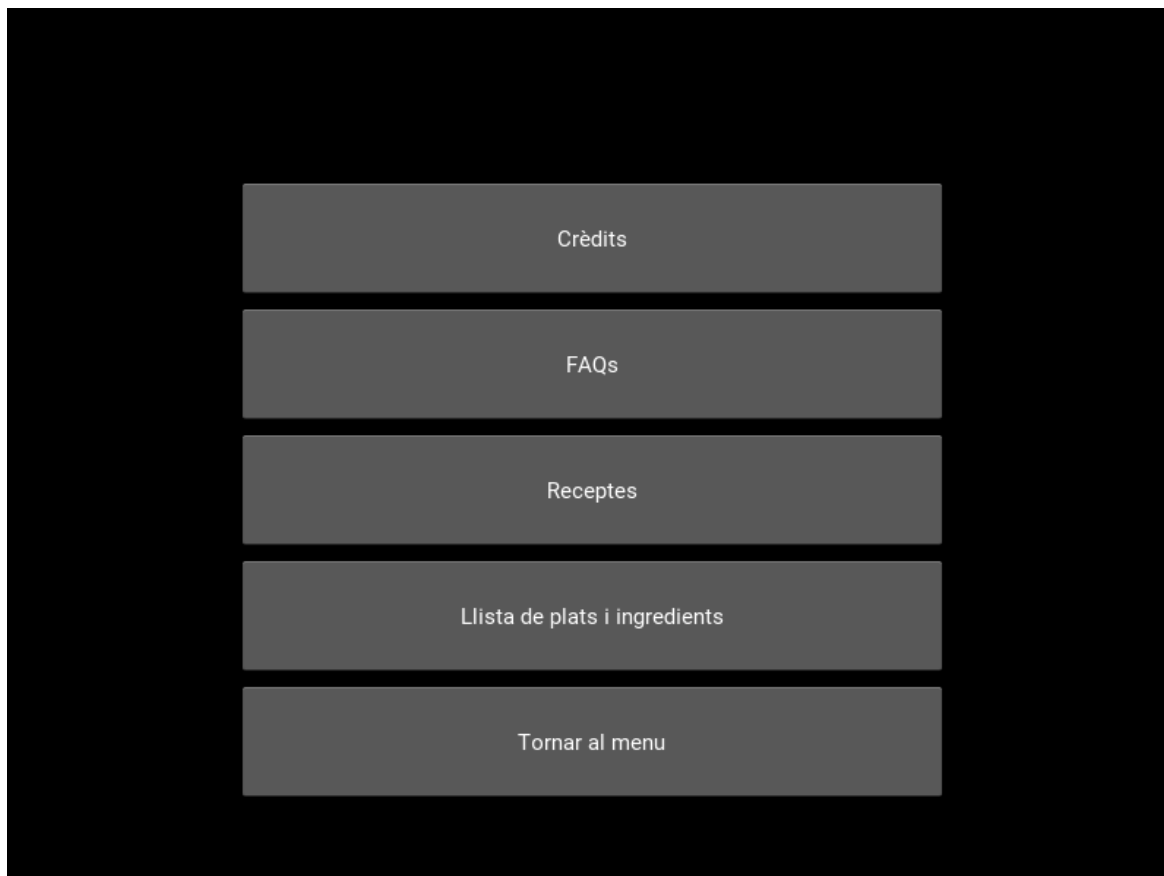
Em vull acabar...

Crèdits, explicacions i receptes



7.4. Informació a l'usuari

En aquesta primera versió de l'aplicació, la informació que s'aporta a l'usuari es troba, bàsicament, dins la secció “Crèdits, explicacions i receptes”. En aquesta secció l'usuari té accés a quatre pantalles com es pot veure a continuació:



- **Crèdits**

Pantalla senzilla que simplement indica el creador i el tutor de l'app i dona un e-mail de contacte per si l'usuari té qualsevol tipus de consulta o dubte.

- **FAQs**

Pantalla senzilla que intenta anticipar-se als dubtes que pugui tenir l'usuari i resoldre'ls. Una vegada més es dona un e-mail de contacte i s'anima a l'usuari a contactar amb el creador si té cap tipus de dubte.

- **Receptes**

Pantalla que conté un botó desplegable on l'usuari pot seleccionar quina és la recepta que vol consultar i se li mostra una explicació de cinc passos amb una imatge del plat. Per a aquesta primera versió només s'han introduït les receptes de sis plats per evitar dedicar la gran quantitat de temps que suposaria crear pantalles de receptes per a les 44 receptes que conté l'app.

- **Llista de plats i ingredients**

Aquesta pantalla únicament conté el llistat de plats i el llistat d'ingredients amb els que treballa l'app i serveix perquè l'usuari pugui consultar si els ingredients que té estan inclosos a la base de dades o no.

Més enllà de la secció específica de “Crèdits, explicacions i receptes”, també hi ha una sèrie de detalls interessants pel que fa a la informació per a l'usuari en les altres seccions.

En primer lloc, les tres seccions principals, “Un sol àpat”, “Múltiples àpats” i “Em vull acabar...” contenen etiquetes per a donar informació a l'usuari. Això és necessari ja que en aquestes pantalles l'usuari ha d'escriure els noms dels ingredients que té i necessita d'un cert feedback per saber si ha introduït correctament el nom de l'ingredient o si aquest ingredient està a la base de dades.

En segon lloc hi ha tota una feina invisible pel que fa a la informació a l'usuari que és present a tot arreu de l'app. S'està parlant, per exemple, de les funcions que serveixen per ensenyar les dades a l'usuari sense utilitzar símbols als quals la gent de fora del món de la programació no estan habituats, com poden ser [], (), ‘ ‘, “ ”. Un altre exemple podrien ser els texts que apareixen a cada pantalla informant de les variables que s'hi visualitzaran o els texts que apareixen a la secció “Múltiples àpats” quan no és possible fer ni un sol plat amb els ingredients disponibles i, per tant, no té cap sentit buscar una segona combinació alternativa.

7.5. Altres

En aquesta secció es comentaran detalls interessants sobre l'aplicació Appat que no s'han pogut comentar en altres seccions.

7.5.1. Funcions de tractament d'errors/excepcions

Les funcions de tractament d'errors/excepcions són el conjunt de funcions que protegeixen al programa quan l'usuari intenta fer quelcom inesperat, aquestes funcions

són part indispensable d'Appat però també de qualsevol altre app o programa d'ordinador. Un cas molt bàsic és el que es produeix quan l'usuari introdueix erròniament el nom d'un ingredient o introdueix el nom d'un ingredient que no és a la base de dades. En absència de funcions de protecció, l'exemple anterior acabaria amb l'app tancant-se a conseqüència d'intentar cridar a un diccionari amb una clau inexistent. És per això que cal crear una sèrie de funcions que eviten aquest tipus d'accions les quals acabarien amb el funcionament de l'app. A continuació es poden veure alguns exemples de les funcions de tractament d'errors i de les lògiques presents al codi Kivy amb aquest mateix objectiu:

- **Avisar a l'usuari quan escriu malament el nom d'un ingredient.** Quan s'introdueix el nom d'un ingredient que no hi és al diccionari, l'usuari rep un avís dient-li que ha escrit un nom erroni i no es continua executant el programa.
- **Possibilitat d'introduir els noms dels ingredients sense distinció entre majúscules i minúscules.** Per evitar que es doni la situació del punt anterior davant d'una situació tant normal com escriure els ingredients amb alguna lletra majúscula, l'aplicació ho treballa tot amb lletres minúscules i aplica el mètode dels strings `string.lower()` a totes les entrades.
- **Avisar a l'usuari quan escriu malament la quantitat d'un ingredient.** Quan s'introdueix una quantitat d'ingredient que no és un nombre, l'usuari rep un avís dient-li que ha escrit malament la quantitat de l'ingredient i no es continua executant el programa.

7.5.2. Implementació en telèfon mòbil

La implementació en telèfons mòbils amb sistema operatiu Android és quelcom que s'ha anat plantejant de del principi del projecte però que finalment s'ha aplaçat per motius diversos. No obstant és una prioritat entre les millores futures i un procés indispensable per al desenvolupament de l'aplicació a nivell professional.

Aquesta implementació es farà amb l'eina Buildozer, oficialment recomanada per Kivy. El Buildozer és una eina que automatitza tot el procés de creació de l'app amb llenguatge Android. Ho fa definint els prerequisits necessaris i construint una apk, el format per a paquets i aplicacions Android, que pot ser automàticament implementada en un dispositiu. Buildozer només funciona en ordinadors de sistema operatiu Linux, un dels motius pels quals s'ha decidit no fer la implementació a telèfon mòbil en la primera versió.

7.5.3. Llengua

L'aplicació Appat està dissenyada i desenvolupada íntegrament en català. No obstant per

a una extensió internacional és necessari que l'aplicació estigui disponible en altres idiomes, principalment anglès i castellà.

Per a poder resoldre aquest tema sense haver de reconstruir des de zero l'aplicació, s'hauria d'aplicar una transformació sistemàtica a tots els texts de la següent manera:

- Modificació dels texts de la interfície d'usuari
 1. Canviar els texts escrits a hores d'ara en el codi Kivy per una variable global de l'aplicació, per exemple `app.etiquetaMenuInicial`.
 2. Crear aquesta variable global de l'aplicació com a llista amb diferents strings per cada idioma, per exemple `["Què voldràs menjar?", "What do you want to eat?", "¿Que querrás comer?"]`.
 3. Crear una primera pantalla a l'inici de l'aplicació on es preguntí en quin idioma es vol treballar i emmagatzemar aquest idioma en una variable `app.index` que correspongui a la posició que té aquest idioma a les llistes de strings. En els exemples anteriors català 0, anglès 1 i castellà 2.
 4. A totes les variables globals de text definides al primer punt, marcar que es cridi la posició `app.index` de la llista, per exemple `app.etiquetaMenuInicial[app.index]`.

- Modificació dels diccionaris de plats i ingredients

La modificació de la llista de plats i ingredients és una tasca molt més laboriosa i és el que fa que a hores d'ara no s'hagi ofert la opció de traduir la aplicació a altres idiomes. Possiblement la millor opció sigui traduir a cada un dels idiomes de l'aplicació el fitxer de text que guarda els noms dels plats i els ingredients. Una alternativa possible seria traduir els noms dels ingredients amb el propi programa per evitar que treballant amb diferents documents es tinguessin diferents nivells d'actualització, és a dir que la llista de plats en un idioma estigui més actualitzada que la d'un altre. No obstant això generaria traduccions incorrectes com, l'amanida russa que en castellà es traduiria com a "ensalada rusa" i no "ensaladilla rusa" com li pertocaria.

7.5.4. Assumpcions i informacions diverses

L'aplicació fa una sèrie d'assumpcions i suposicions que, tot i ser molt òbvies, és important enumerar. A dins de l'aplicació aquestes assumpcions i informacions es poden consultar a la secció "FAQs", dins del menú "Crèdits, explicacions i receptes" i es poden

veure en la figura que es presenta a continuació:

La secció de FAQs (Frequently Asked Questions) pretén respondre als dubtes ocasionals que pugui tenir l'usuari. Si els vostres dubtes no estan resposts en aquesta secció, podeu escriure un e-mail a appat.info@gmail.com i us respondrem tant aviat com sigui possible

L'aplicació Appat assumeix que l'usuari té aigua potable, oli, sal i els estris adequats per a cuinar els plats necessaris.

L'aplicació treballa sense accents en els noms dels plats i els ingredients degut a problemes del Python 2.7.11 amb windows. Això s'arreglarà tant bon punt s'actualitzi la aplicació a la versió 3.#. A més, per a poder permetre a l'usuari escriure el text sense tenir en compte majúscules i minúscules, tots aquests noms estan en minúscules i així es veuran quan l'aplicació retorni els noms dels plats

Per facilitar la feina a l'usuari i personalitzar l'app a les seves exigències, l'aplicació Appat rep les quantitats d'aliments en forma de racions, les quals seran aportades per l'usuari jutjant si la quantitat que té. S'entén per ració la quantitat d'un ingredient principal suficient per a fer un plat. A tall d'exemple, una ració de pasta estaria entre 90 i 130 grams

Tornar enrere

No es parlarà amb més detall sobre aquests punts perquè ja es tracten en altres seccions de la memòria.

8. Base de dades

La base de dades de l'aplicació Appat està creada pel mateix autor i permet donar uns resultats satisfactoris a l'aplicació tot i ser molt bàsica. Per dedicar-li l'atenció que es mereix, s'hauria pogut desenvolupar un TFG sencer sobre la base de dades i la seva relació amb el codi mentre una altra persona s'ocupés de la interfície d'usuari i l'heurística.

L'actual base de dades consta d'un nombre elevat de plats, 44, i només presenta un defecte major i és que, com a mal menor, hi ha plats on s'han hagut de definir ingredients principals que no haurien de ser-ho. Això passa perquè hi ha plats que requereixen d'un ingredient principal i alguns dels seus acompanyaments independentment de quins i un exemple en podria ser l'amanida de pasta. Per a una amanida de pasta, la pasta és l'únic ingredient indispensable però és insuficient per a cuinar el plat, doncs fan falta un mínim d'acompanyaments però cap d'ells és indispensable. Per exemple l'amanida de pasta pot portar pasta, olives verdes, Frankfurt i pastanaga o portar pasta, tonyina, blat de moro i olives negres.

Davant d'aquest problema, s'ha optat per aquesta solució de definir alguns dels acompanyaments, generalment el blat de moro, com a ingredients principals. En una propera versió es podria arreglar aquesta problemàtica, creant una nova classificació pels ingredients que indiqués si són d'aquesta categoria de "semiacompanyaments" i fent que el programa donés per cuinable el plat sempre que tingués, com a mínim, 2 o 3 semiacompanyaments. És cert que s'hauria pogut ignorar aquesta problemàtica i definir només com a ingredient principal la pasta però s'ha preferit tenir falsos negatius, provocats per la solució actual, que falsos positius, provocats per la pasta com a únic ingredient principal. En tot cas la solució d'aquest problema és prioritària per a les versions futures.

8.1. Implementació

La base de dades està creada a partir d'un document de Microsoft Excel. El document original conté el nom d'un plat a la primera columna i, a partir d'aquí, cada columna conté el nom d'un ingredient i la quantitat necessària per a la recepta, en racions, separada per un espai i entre parèntesi. Per exemple: arros (2). Aquest document es guarda en format .csv, una opció de Microsoft Excel que permet guardar el document com a fitxer de text on el contingut de cada cel·la està separat per una coma del de la cel·la prèvia i la posterior. Finalment, el codi d'Appat conté una funció capaç de crear el diccionari de plats a ingredients a partir d'aquest fitxer .txt separat per comes. A continuació es mostra un

extracte del text que conté aquest fitxer .txt.

```

amanida d'arros,arros (1),blat de moro (1),**llenties**,**tonyina**,**olives**,
amanida de cigrons,cigrons (1),blat de moro (1),**olives**,**ou**,**tomaquet**,**tonyina**,
amanida de pasta,pasta (1),blat de moro (1),**tonyina**,**tomaquet**,**mozzarella**,**olives**,
amanida de patata,patata (1),tonyina (1),**pastanaga**,**frankfurt**,**blat de moro**,**olives**,
**amanida verda**,enciam (1),**endivies**,**pastanaga**,**blat de moro**,**tomaquet**,**ou**,**ceba**,**rostes**
**amanida cesar**,enciam (1),pollastre (1),salsa cesar (1),**rostes**,**endivies**,**tomaquet**,
arros 3 delícies,arros (1),gambes (1),**truita**,**baco**,**pesols**,**pastanaga**,
arros al curri,arros (1),curri (1),**crema de llet**,**ceba**,**pollastre**,
arros amb salsa de tomaquet ,arros (1),salsa de tomaquet (1),**ou**,
arros amb soja,arros (1),salsa de soja (1),**pollastre**,
arros bullit,arros (1),**all**,**peix**,
botifarra,botifarra (1),**bolets**,**seques**,
carn a la planxa,carn vermella (1),**salsa de tomaquet**,**curri**,^amanida verda^,
carn arrebossada,carn vermella (1),ou (1),pa ratllat (1),**enciam**,**endivies**,
salsitxes,salsitxes (1),**seques**,**arros**,**patata**,**ceba**,^amanida verda^,
hamburguesa,carn picada (1),**salsa de tomaquet**,**formatge**,**patata**,**pa**,**ceba**,**enciam**
llenties,llenties (1),**ceba**,**embotit**,**arros**,
pasta amb salsa de soja,pasta (1),salsa de soja (1),ceba (1),**pastanaga**,
pasta amb salsa de tomaquet,pasta (1),salsa de tomaquet (1),**formatge**,
pasta carbonara,pasta (1),crema de llet (1),**baco**,**bolets**,**formatge**,**ceba**,
**patates fregides**,patata (1),**ketxup**,**maionesa**,
peix a la planxa,peix (1),^amanida verda^,
peix al forn,peix (1),**patates**,**ceba**,**tomaquet**,
peix arrebossat,peix (1),farina (1),**maizena**,**ou**,
peix bullit,peix (1),**farigola**,**all**,
pesols amb baco,pesols (1),baco (1),**ceba**,
pollastre a la planxa,pollastre (1),**salsa de tomaquet**,**curri**,^amanida verda^,
pollastre al forn,pollastre (1),**bolets**,**ceba**,**especies**,^amanida verda^,
pollastre amb soja,pollastre (1),salsa de soja (1),**sesam**,^amanida verda^,
pollastre arrebossat,pollastre (1),ou (1),pa ratllat (1),**salsa de tomaquet**,**patates**,^amanida verda^,
rissoto de bolets,arros (1),bolets (1),crema de llet (1),**formatge**,
rissoto de verdures,arros (1),ceba (1),**carbasso**,**bolets**,**crema de llet**,**formatge**,
truita de patates,ou (1),patata (1),**ceba**,**all**,
verdures saltejades,ceba (1),**carxofa**,**albergínia**,**pebrot vermell**,**pebrot verd**,**carbasso**,
verdures saltejades amb pa ratllat i oli,pa ratllat (1),ceba (1),**pebrot verd**,**pebrot vermell**,**bolets**,**embotit**,**carxofa**

```

Degut a problemes en l'aplicació de Python en Windows, els noms dels ingredients estan escrits sense accents. Aquest fet es deu a que, independentment de definir “ -*- coding: utf-8 -*-” al principi del document, el Python 2.7.11 per Windows emmagatzema les lletres amb accents com a combinacions de caràcters que després Kivy no reconeix de la mateixa manera. Un exemple d'emmagatzemament estrany seria el següent:

```

>>> t='Arròs'
>>> t
'Arr\x95s'

```

Això pot semblar contradictori amb el fet que l'aplicació utilitza accents en les etiquetes d'informació, però no ho és ja que els texts d'etiquetes i botons de Kivy mai són processats per cap programa i es mostren tal qual s'indiquen.

8.2. Contingut

La versió definitiva de la base de dades ha acabat contenint la mateixa informació que s'esperava quan es va definir la base de dades, veure secció 6.1, i aquestes informacions permeten:

- **Determinar si els ingredients són ingredients principals o acompanyaments.**
Això es fa escrivint primer els plats principals i, després, escrivint els ingredients d'acompanyament entre asteriscs, per exemple ****Ceba****. Això permet a l'aplicació trobar la millor combinació d'àpats possibles utilitzant només els ingredients indispensables per evitar malgastar ingredients importants per a altres plats com a acompanyaments d'un primer.
- **Determinar la quantitat necessària d'un ingredient per a poder cuinar un plat determinat.**

Tal i com s'ha comentat anteriorment, la base de dades determina la quantitat de racions d'un ingredient necessàries per a poder cuinar un plat determinat definint-la entre parèntesi després del nom de l'ingredient.

- **Definir quins són els plats-acompanyaments**

Els plats-acompanyaments són plats que poden ser plats per si sols o actuar com a acompanyaments d'altres plats i és el cas de les amanides. Aquests plats-acompanyaments estan definits entre ******* de la mateixa manera que els acompanyaments quan se'ls defineix com a plats i entre **^^** quan se'ls defineix com a acompanyaments d'un plat.

8.3. Base de dades en format diccionari

Un cop és llegida per l'aplicació aquesta base de dades, emmagatzemada en un fitxer de text, es converteix en dos diccionaris. El diccionari de plats a ingredients i el diccionari d'ingredients a plats.

8.3.1. Diccionari plats a ingredients

Aquest diccionari es crea amb la funció "csvDiccionari", la qual rep com a entrada el nom del fitxer de text i retorna aquest primer diccionari. El diccionari de plats a ingredients conté com a claus els noms dels plats definits a l'arxiu de text i, com a valors d'aquestes claus una llista on cada terme d'aquesta llista és el nom d'un ingredient separat per un espai i la quantitat de racions necessàries per a aquest plat entre parèntesi. Si els ingredients són acompanyaments van entre dos asteriscs i no contenen quantitat ja que queda a elecció de l'usuari escollir quant en posa i si li sobra estoc d'aquest ingredient per fer-ho. Aquest diccionari té el següent aspecte:

{'angules': ['angules (1)', 'gambes (1)', '**pebre vermell**', '**Ceba**'], 'truita de patates':

['ou (1)', 'patata (1)', '**Ceba**', '**all**']...}

8.3.2. Diccionari ingredients a plats

Aquest diccionari prové de la funció “invertirAmbQuantitats” que a partir del diccionari de plats a ingredients crea una mena de diccionari invers. És sabut que no es pot invertir un diccionari i que, purament, el diccionari invers voldria dir que la llista dels ingredients cridaria a la clau. El diccionari d'ingredients a plats conté el nom d'un ingredient i, com a valors d'aquesta clau, una llista amb els plats on apareix aquest ingredient. Aquest diccionari té el següent aspecte:

{'angules':['angules'], 'maionesa':['patates fregides'], 'all':['truita de patates', 'peix bullit', 'arros bullit']...}

9. Proves

Una vegada l'algoritme de cerca i la llista d'ingredients han estat en versió definitiva, s'ha procedit a una sèrie de proves per determinar si les prestacions són les desitjades.

9.1. Criteri

El criteri de les proves és de vital importància ja que permet determinar si el funcionament del programa és satisfactori o no. Com és evident, el criteri de les proves cal definir-lo abans d'haver posat a prova l'aplicació.

Des del punt de vista de l'usuari, la qualitat es percep des de 3 àmbits. El temps de realització, la qualitat de la realització i la profunditat de la realització, entenent com a qualitat el fet de rebre una solució satisfactòria i com a profunditat el fet de rebre diverses solucions possibles. Davant aquests factors es van definir els següents criteris:

- Temps d'espera per part de l'usuari < 2 segons.

En absència d'un indicador de progrés del procés de càlcul, l'usuari pot entendre com a fallada en el funcionament de l'app un temps d'espera mitjà o gran. Per tant, es limita el temps d'espera per part de l'usuari a un màxim de 2 segons.

- 0% de falsos positius.

Una aplicació d'aquestes característiques no es pot permetre entregar cap resultat que no sigui realment possible ja que això destrossaria la confiança de l'usuari en el producte. Per tant, no s'acceptarà cap programa que retorni falsos positius, és a dir, solucions en què el programa afirmi que es pot cuinar un plat i realment no es disposi dels ingredients necessaris. Aquest criteri també s'aplica a l'assignació d'acompanyaments.

- 0% de sortides del programa

De la mateixa manera que un programa desenvolupat amb Python, una aplicació desenvolupada amb Kivy es tanca immediatament si detecta alguna incoherència o acció impossible a nivell de codi. Exemples d'accions impossibles serien utilitzar un índex fora del rang, intentar definir una propietat que no correspon a aquell tipus d'estructura de dades...

Les sortides del programa són el més greu dels errors ja que l'usuari és plenament

conscient de què el programa ha fallat. Per molt que sigui l'usuari qui pugui haver desencadenat la fallada, el responsable en última instància és el programador ja que s'ha d'anticipar a aquest tipus de fallades i, per tant, no s'admetrà una app on en alguna de les seves funcions, el programa es tanqui a causa d'un error.

- Nombre mínim de falsos negatius per a combinatòries senzilles.

De la mateixa manera que l'aplicació no es pot permetre entregar cap resultat de fals positiu, tampoc pot permetre's que l'usuari detecti gaires resultats de fals negatiu. S'entén com a fals negatiu aquella situació on el programa afirma que no existeix solució possible però realment n'hi ha. Conseqüentment, no s'acceptarà cap programa que, davant menys de tres àpats (inclòs 3) per a dos comensals retorni falsos negatius. Aquest criteri també s'aplicarà a l'assignació d'acompanyaments, on no serà acceptat que un ingredient que queda definit com a sobrant, fos afegible a cap dels plats del menú.

- 0% de falsos positius per als resultats alternatius

La profunditat de les proves s'avalua d'una manera menys estricta i no es consideren els falsos negatius de les solucions alternatives perquè no són una funcionalitat principal. No obstant l'existència de falsos positius, situacions en les quals l'aplicació proposa quelcom impossible, segueix sent totalment inacceptable.

9.2. Paràmetres de les proves

Les proves s'han dissenyat seguint un procés invers. En primer lloc, s'han determinat una sèrie de menús per un cert nombre de persones i un cert nombre d'àpats, en segon lloc, s'han comptabilitzat els ingredients necessaris per a cada àpat i, finalment, s'ha executat el programa amb els ingredients exactes i afegint-hi ingredients extres per mirar d'enganyar al programa. Aquest procés s'ha dut a terme per a les seccions "Un sol àpat" i "Múltiples àpats" i s'ha personalitzat a les necessitats d'aquesta. No s'han fet proves per a la secció "Em vull acabar..." perquè té una lògica tant senzilla que no dona marge d'error.

9.2.1. Un sol àpat

Per a les proves de la secció "Un sol àpat" es van dissenyar onze proves amb diferents característiques buscant explotar les possibles debilitats de l'aplicació. A la següent taula es poden veure aquestes proves amb els seus respectius motius. És important recordar que l'aplicació treballa sense accents, així doncs les proves, que es mostren a continuació, també s'han dissenyat així.

| Prova | Plat | | Ingredients principals | Ingredients secundaris | Ingredients trampa | Característiques prova |
|-------|----------------------|---|---------------------------|--|-------------------------------------|---|
| 1 | arros delícies | 3 | arros | gambes, truita, pesols | - | Prova bàsica. |
| 2 | pesols amb baco | | pesols, baco | ceba | - | S'afegeix un segon ingredient principal. |
| 3 | pollastre amb soja | | pollastre, salsa de soja | sesam, enciam | - | Es busca forçar un plat-ingredient (amanida verda) com a acompanyament. |
| 4 | amanida d'arros | | arros, blat de moro | llenties | Enciam | S'afegeix un ingredient trampa. |
| 5 | amanida verda | | enciam | Endivies, blat de moro, tomaquet, ou, ceba | pollastre | Es busca tenir un plat-ingredient (amanida verda) com a plat principal. |
| 6 | patates fregides | | patata | ketxup, maionesa | | S'introdueixen primer els acompanyaments al programa. |
| 7 | arros al curri | | arros, curri | crema de llet, pollastre | pollastre, salsa de tomaquet, curri | Els ingredients trampa són suficients per a un altre plat sencer amb acompanyaments (pollastre a la planxa). |
| 8 | pa amb embotit | | pa, embotit | | - | No es proposen acompanyaments. |
| 9 | verdures saltejades | | | carxofa, albergínia, pebrot vermell, pebrot verd, carbasso | - | Es fa un plat que només té acompanyaments sense ingredients trampa. |
| 10 | verdures saltejades | | | carxofa, albergínia, pebrot vermell, pebrot verd, carbasso | ou, pollastre | Es fa un plat que només té acompanyaments amb ingredients trampa. Els ingredients trampa permeten fer plats que comparteixen acompanyaments amb el plat principal (ou remenat). |
| 11 | pollastre arrebossat | | pollastre, ou, pa ratllat | enciam, pastanaga | - | Es busca forçar un plat-acompanyament com a acompanyament (amanida verda) i que aquest tingui acompanyaments propis. |

9.2.2. Múltiples àpats

Les proves de la secció múltiples àpats són necessàriament més complexes que les de la secció “Un sol àpat”. Més enllà del fet de considerar quantitats, múltiples àpats i múltiples comensals, la funció “Múltiples àpats” és la joia de la corona d’Appat, la funció que la converteix en una aplicació diferent i única. Conseqüentment cal sotmetre-la a prova amb més profunditat. Es va creure necessari arribar fins a les 23 proves per tal de poder examinar totes els possibles errors de l’aplicació. A continuació se’n pot veure el disseny:

| Prova | Plat principal | Ingredients principals | Ingredients secundaris | Ingredients trampa | Comensals | Àpats | Característiques prova |
|-------|-------------------------------------|---|-------------------------------------|----------------------------|-----------|-------|---|
| 1 | arros delícies | 3 arros (1) | gambes (1), truita (1), pesols (1) | | 1 | 1 | Prova bàsica. |
| 2 | pesols amb baco | (1), pesols baco (1) | ceba (1) | | 1 | 1 | S'afegeix un segon ingredient principal. |
| 3 | pasta amb salsa de tomaquet | (1), pasta salsa de tomaquet (1) | formatge (1) | pesols (1), pinya (1) | 1 | 1 | S'hi afegeixen ingredients trampa. |
| 4 | amanida cesar | (2), enciam pollastre (2) | endivies (2), rostes (2) | | 2 | 1 | S'hi afegeix un segon comensal, quantitat exacta de tots els ingredients. |
| 5 | Pollastre al forn | (2) pollastre | (1), bolets cebra (1) | | 2 | 1 | Es va curt d'acompanyaments. |
| 6 | hamburguesa | (3) carn picada | Salsa de tomaquet (1), formatge (1) | arros (2), patata(1) | 2 | 1 | Es va sobrat d'ingredient principal. |
| 7 | botifarra, truita de patates | (3), botifarra(3), ou (3), patata(3) | (1), seques (1), cebra (1) | tomaquet (1), poma (3) | 3 | 2 | Es fan múltiples àpats. |
| 8 | rissoto de bolets, carn a la planxa | (3), arros bolets (3), crema de llet (3), carn vermella (3) | (1) Curri | (2), baco (2), cebra (5) | 3 | 2 | Un dels múltiples plats té múltiples ingredients principals. |
| 9 | botifarra, truita de patates | (4), botifarra(4), ou (2), patata(3) | (1), seques (1), cebra (1) | (1), Pesols (1), Pinya (1) | 3 | 2 | Algun dels ingredients principals hi és en una quantitat insuficient. |

| | | | | | | | |
|----|--|--|--|------------------------------------|---|---|--|
| 10 | Ous amb baco, pollastre a la planxa | ou (6), pollastre (3), baco (3) | Salsa de tomaquet (3), pa (5) | peix(4), gambes(2), pa ratllat (3) | 3 | 2 | S'utilitza un ingredient del qual se'n necessita doble ració per a fer un plat. |
| 11 | truita a la francesa, ou remenat | ou (16), | mel (15), bolets (2), pebrot vermell (1) | baco (4) | 4 | 2 | Els plats comparteixen un dels ingredients principals. |
| 12 | carn arrebossada, amanida de cigrons | carn vermella (3), ou(3), pa ratllat (3), cigrons (3) | blat de moro (5) | pastanaga (3) | 3 | 2 | S'utilitza i esgota un ingredient principal que és acompanyament al següent plat. |
| 13 | salsitxes, peix bullit | salsitxes (4), enciam (3), peix(5) | arros (1), seques (2), pastanaga (3), farigola (3) | | 3 | 2 | S'utilitza un plat-ingredient com a acompanyament. |
| 14 | amanida verda, pasta amb salsa de soja | pasta (3), salsa de soja (3), ceba (6), enciam (5) | Pastanaga (2) | | 3 | 2 | S'utilitza un plat-ingredient com a plat. |
| 15 | pollastre al forn, amanida verda | Enciam (4), pollastre(3) | Pastanaga (2), bolets (1) | | 3 | 2 | S'utilitza un plat-ingredient com a plat i com a acompanyament. |
| 16 | arros bullit, salsitxes | arros (4), salsitxes (3), | peix (2) | | 3 | 2 | S'utilitza sense esgotar un ingredient principal que és acompanyament al següent plat. |
| 17 | amanida de cigrons, amanida de patata | patata (3), cigrons (3) | blat de moro (5), olives (2), tonyina (1) | | 3 | 2 | Acompanyament comú. |
| 18 | rissoto de verdures | arros(1), arros (1) | | | 2 | 1 | S'introdueix primer una part de l'ingredient i després la resta. |
| 19 | Pasta carbonara | Pasta (1), pasta (2), crema de llet (2), crema de llet (1) | Ceba (2) | | 3 | 1 | S'introdueixen diverses combinacions de l'ingredient. |

| | | | | | | |
|----|--|---|----------------------------------|---|---|---|
| 20 | rissoto de verdures, carn arrebossada, peix bullit | arros (3), pollastre(3), ou(3), pa ratllat(3), peix (3) | Ceba (3), enciam (3) | 3 | 3 | Tres plats. |
| 21 | Arros amb soja | Arros (2), salsa de soja (2) | Pollastre (0) | 2 | 1 | Es dona zero d'alguna quantitat d'ingredient acompanyament. |
| 22 | verdures saltejades amb pa ratllat... | pa ratllat (2) | Ceba (0), bolets (1), embotit(2) | 1 | 2 | Es força menjar el mateix en dos àpats seguits. Atenció a veure què passa amb els acompanyaments. |
| 23 | verdures saltejades amb pa ratllat... | pa ratllat (1) | | 1 | 1 | Plat sense acompanyaments ni ingredients trampa. |

És important denotar que la base de dades de plats i ingredients s'ha modificat després de les proves per a millorar-la. Això, no obstant, no és cap conseqüència de les proves ni les invalida ja que les proves no són per testejar la base de dades sinó per testejar els algorismes de l'app.

9.3. Execució i resultats de les proves

L'execució de les proves es podria haver fet dissenyant un programa expressament per això, però es va decidir que el la comprovació manual prova per prova era més adient per a les primeres execucions d'aquestes. Això es deu a diversos fenòmens, entre ells fet que a vegades l'aplicació no suggereix els plats que s'esperava que suggerís perquè hi ha altres combinacions possibles. Només executant el programa a mà es poden veure aquests problemes i modificar les quantitats d'ingredients sobre la marxa per poder forçar la situació que es vol observar. Aquest procés també va ajudar a afegir alguna situació concreta a les proves en la qual no s'havia pensat durant el disseny d'aquestes.

Com a observació general de les proves, totes s'han resolt a l'instant sense mostrar el mínim símptoma d'esforç per part de l'ordinador. Això és bo ja que es desitgen programes ràpids però hom podria argumentar que aleshores potser valdria la pena fer una cerca més exhaustiva ja que el programa segueix sent ràpid. Un contra argument seria recordar que la base de dades és petita i experimental a hores d'ara però que amb l'aplicació plenament desenvolupada seria més gran i, per tant, requeriria iteracions més llargues al programa. També es podria contra argumentar que si es vol introduir l'aplicació en

dispositius mòbil aquests no tenen tanta potència de càlcul com els ordinadors.

Després d'una primera ronda de proves en la qual els resultats van ser bastant dolents, s'han introduït algunes modificacions al programa i s'ha fet una segona ronda de proves per comprovar l'efectivitat d'aquestes modificacions. A continuació s'analitzen les proves per a cada secció, si es desitja veure els resultats exactes de les proves es troben a les taules 1 i 2 de l'annex.

9.3.1. Un sol àpat

La primera ronda de proves va ser bastant descoratjadora ja que només cinc de les onze proves van resultar completament satisfactòries. Entre els motius principals de les fallades es van identificar els següents:

- La funció `stringueja`, una funció dedicada a retornar el resultat en forma de string eliminant símbols típics del món de la programació com `[]`, `()`, `" "`, no funcionava correctament. S'ha modificat la funció des de zero i ara es basa en el mètode `string.replace()` dels strings de Python.
- L'element llistat immediatament darrere d'un plat-acompanyament no entrava com a acompanyament encara que hagués de fer-ho. Aquest problema s'ha resolt modificant un índex de la funció `actualitzaLlistaIngredients`.
- La segona alternativa de plat que més aliments esgotava a vegades quedava en blanc però, en canvi, hi havia una tercera alternativa amb acompanyaments, cosa que no té sentit si se suposa que la segona utilitza més ingredients que la tercera. Això es devia al fet que alguns plats requereixen de més d'un ingredient principal, cosa que generava que encara que no tinguessin acompanyaments assignats, utilitzessin més ingredients que altres però aleshores quedaven en blanc perquè no se'ls podia assignar cap acompanyament. Això s'ha resolt afegint una lògica en què si una posició queda buida, cosa que passa quan no té ingredients per a assignar, s'ensenyi el plat que li pertocaria encara que sigui sense ingredients acompanyament.

Un cop identificades les funcions problemàtiques, s'han resolt i s'ha fet una segona ronda de proves molt més satisfactòries on només una de les onze proves donava resultats una mica diferents als esperats però tampoc estava malament. Aquest resultat diferent retornava un ordre òptim dels plats no era el que aparentment hauria de ser, però s'ha considerat un error negligible ja que l'usuari rep la mateixa informació. De totes maneres es pot corregir per a la següent versió.

Després de les dues rondes de proves i la substancial millora de la funcionalitat entre una i l'altra, s'ha determinat que el funcionament de la secció "Un sol àpat" és satisfactori per a la primera versió de l'aplicació.

9.3.2. Múltiples àpats

De la mateixa manera que per a la funció "Un sol àpat", la primera ronda de proves va ser bastant descoratjadora tot i que va anar una mica millor que la d'aquest primer apartat. Es van donar 14/24 proves correctes aquí per un 5/11 allà i entre els motius principals de les fallades es van identificar els següents:

- Faltava reinicialitzar un comptador a la funció `actualitzaLlistaIngredientsMultiple`. Això provocava que la funció només pogués actualitzar la quantitat del primer ingredient de la llista.
- La funció `assignaAcompanyaments` tenia un funcionament recursiu innecessari que donava problemes en algunes situacions.
- L'assignació d'ingredients de la combinació alternativa quedava com a llista buida. Aquest problema es devia a que la variable global de la llista de plats i acompanyaments no estava definida com a `ListProperty()` com sí que ho estava la primera.

Un cop modificat el programa s'ha fet una segona ronda de les proves on s'han visualitzat uns resultats molt més acceptables però que no són tant bons com els finals de la funció "Un sol àpat".

En un dels casos negatius, l'heurística falla i la primera combinació retorna un fals negatiu. S'afirma que no és possible cuinar aquest plat amb els ingredients dels que es disposa però la pròpia combinació alternativa ho desmenteix oferint una combinació possible. No deixa de ser un error resolt per la pròpia aplicació gràcies a la combinació alternativa però és quelcom que s'hauria de millorar de cara a una segona versió de l'app.

L'altre error persistent després de la revisió és la no assignació d'acompanyaments als plats-acompanyaments quan aquests actuen com a acompanyaments. Aquest error no es produeix sempre i alguns d'aquests casos s'han solucionat quan s'ha fet no recursiva la funció `assignaAcompanyaments`. Es considera que això s'hauria de corregir de cara a una segona versió del programa però no es considera una fallada crítica ja que no retorna cap fals positiu i, a més, es tracta d'una fallada en els acompanyaments d'un acompanyament, no d'un plat. Un exemple d'aquest error és que quan l'amanida verda actua com a acompanyament i sobra pastanaga, no s'assigna la pastanaga a l'amanida

verda.

Després de la segona ronda de proves es pot considerar que la secció Múltiples àpats és apta però serà necessari dedicar-hi esforç de cara a una segona versió perquè aquesta funció és l'ànima d'Appat i no acaba de funcionar tant bé com es voldria.

10. Planificació temporal i costs

10.1. Planificació temporal

Per a desenvolupar l'aplicació Appat han estat necessàries aproximadament unes 250 hores de treball directe entre definició i desenvolupament de l'aplicació i redacció de la memòria. A més, prèviament ha calgut invertir unes 30 hores d'aprenentatge de Kivy per a poder desenvolupar l'aplicació.

10.2. Costs

10.2.1. Propietat intel·lectual

A l'hora de dissenyar quelcom nou, és precís posar atenció en la protecció de la propietat intel·lectual. Segons la pàgina web del ministeri d'indústria, en algun moment o altre serà necessari protegir jurídicament la propietat intel·lectual i convé fer-ho el més aviat possible, fins i tot abans de donar a conèixer la idea.

Existeixen diverses maneres jurídiques de protegir una propietat intel·lectual més enllà de la coneguda patent. L'efectivitat de cada una d'elles és variable en funció de la propietat intel·lectual i la seva elecció depèn estrictament de coneixements jurídics que queden fora de l'àmbit de l'enginyeria. És per això que el ministeri defineix com a essencial l'assessorament per part d'un agent o advocat especialitzat en propietat intel·lectual. Partint d'aquesta necessitat d'assessorament, s'ha fet una cerca de diverses companyies que es podrien ocupar d'aquest tema si es desitgés tirar endavant el projecte Appat a un nivell comercial. Els noms de les companyies i les pàgines web es troben a la bibliografia.

A nivell de costs, totes les companyies estan disposades a fer un pressupost gratuït a partir d'una certa informació bàsica sobre el projecte, de totes maneres aquest no és el camp d'estudi del projecte i no s'ha demanat cap pressupost per no fer perdre el temps a aquells treballadors. Per als càlculs de costs, doncs, s'han pres com a referència els valors que aporta l'empresa SpeedPatent, la qual anuncia que per 165€ pot proporcionar un registre de marca a nivell espanyol amb tots els serveis que comporta i per 1150€ pot fer-ho a nivell europeu. Es consideraran els costs europeus per a calcular el cost total del projecte.

10.2.2. Servidors i Google play

Després de cercar diverses pàgines web i articles sobre servidors, [17] i [18], s'ha estimat que el cost d'un servidor dedicat estarà al voltant dels 1200€ incloent-hi el manteniment a un any. De totes maneres per a l'aplicació Appat un servidor és una eina massa potent perquè l'únic que es necessita és poder tenir els arxius de l'aplicació i la base de dades perquè els usuaris puguin descarregar-s'ho. Així doncs és possible que un núvol fos suficient però, davant del dubte i com que això queda fora de l'estudi, s'assumiran els 1200€ del servidor.

Segons emprendedores.es, [19], el cost de pujar una aplicació a l'App Store l'any 2012 estava entre 99 i 300 dollars, de manera que es defineix un valor mig de 130€. D'altra banda, per a pujar una aplicació a Google play només fa falta donar-se d'alta com a desenvolupador d'apps a Google pel mòdic preu de 25 dollars. Malauradament Google es quedarà incondicionalment amb el 30% dels beneficis que generi l'app.

10.2.3. Mà d'obra

Per a desenvolupar l'aplicació Appat han estat necessàries 250 hores de treball directe, de les quals 75h es podrien atribuir a un enginyer de software sènior, que correspondrien a la definició del projecte, el traçar les línies mestres de l'app i una sèrie de revisions finals, i 175h es podrien atribuir a un enginyer de software junior. Segons PayScale, de mitjana, un enginyer de software sènior cobra 36013 € bruts/any a l'estat espanyol mentre que un enginyer junior cobra, de mitjana, 30144 € bruts/any. Com que la pàgina no especifica ni quantes vacances ni quantes hores a la setmana treballen aquests empleats tipus, s'assumirà que tenen 25 dies de vacances a l'any i que treballen 8 hores diàries els 5 dies de la setmana per a poder fer el càlcul de quant cobren per hora.

365 dies/any dels quals 250 són laborables. Si fan 25 dies de vacances els enginyers treballen $225 \text{ dies} \times 8 \text{ h/dia} = 1800 \text{ h/any}$. Tenint aquest valor, el salari per hora d'un enginyer de software sènior és de 20€/hora i el d'un enginyer de software junior és de 16,8€/hora, valor que s'arrodoneix a 17€/hora. Calculant el salari d'aquesta manera se'ls està pagant la part proporcional de vacances a cada hora.

Per últim, una implementació professional de l'aplicació Appat requeriria dels serveis d'un dissenyador de pàgines web durant unes 25 hores per tal de dissenyar una interfície d'usuari atractiva per l'usuari. També segons PayScale, el dissenyador cobra 20000€ anuals de mitjana, de manera que el seu salari és de 11,1 €/hora, valor que s'arrodoneix a 11 €/hora.

10.2.4. Costs totals

Amb les dades que s'han anat definint als anteriors apartats, es pot estipular que el projecte Appat té un cost d'uns 7500-8000€ depenent de si l'enginyer júnior aprèn el llenguatge Kivy en hores de feina o ho fa per compte propi. El procés sencer de càlcul es pot veure a la taula que es presenta a continuació:

| Font del cost | Cost fix (€) | Cost/hora (€/hora) | Nombre d'hores | Cost total (€) |
|------------------------------------|--------------|--------------------|----------------|----------------|
| Registre de marca a nivell europeu | 1150 | | | 1150 |
| Servidors | 1200 | | | 1200 |
| Pujar l'app a Google Play | | | | 25 |
| Pujar l'app a l'App Store | | | | 130 |
| Hores enginyer Jr. | | 17 | 175 (+30) | 2975 (+510) |
| Hores enginyer Sr. | | 20 | 75 | 1500 |
| Hores dissenyador pàgines web | | 11 | 25 | 275 |
| Total arrodonit | | | | 7500 (8000) |

11. Impacte mediambiental

En tractar-se d'una aplicació per a ordinador, Appat no té un impacte mediambiental directe perceptible, doncs el seu desenvolupament i distribució no té cap cost material. No obstant, s'espera que l'impacte mediambiental indirecte de l'aplicació sigui molt positiu depenent, bàsicament, del nivell d'extensió que assoleixi aquesta. Val la pena recordar que l'objectiu principal d'Appat és combatre el malbaratament alimentari i això repercutirà directament en una reducció de l'impacte mediambiental mundial, "Si som capaços d'organitzar millor els nostres àpats, se'ns farà malbé menjar".

A nivell mediambiental, el malbaratament alimentari és el més gran dels despropòsits. Es tracta d'aliments que han consumit grans quantitats de recursos naturals per a la seva producció/fabricació que no només no s'aprofiten sinó que causen un impacte negatiu amb la seva degradació. Per exemple es necessiten 15000 L d'aigua i 6,5 kg de blat per a produir un sol quilo de carn, per no parlar de l'impacte directe de la seva producció, els purins. Es cert que l'aigua no s'esgota sinó que segueix un cicle i que un kg de carn llençat, un cop degradat, adobarà els camps de la mateixa manera que els purins. No obstant l'impacte segueix sent negatiu perquè grans quantitats de menjar, o purins no tractats, llençades en abocadors, o camps respectivament, tenen un impacte devastador sobre els ecosistemes locals ja que els desequilibren per complet.

Un cop coneguda la importància del malbaratament alimentari, és important ser conscients de la mida del problema. A nivell local, segons informació oficial de l'Àrea Metropolitana de Barcelona, a Catalunya es llencen 35 kg de menjar per habitant i any, cosa que equival a unes 262 tones de residus anuals. A nivell global, però, és on les dades adquireixen dimensions esgarrioses ja que, segons dades de la FAO (Food and agriculture Organization of the United nations), 1/3 del menjar produït arreu del món es llença o es fa malbé, assolint els 1300 milions de tones anuals. És destacable el fet que aquest problema es produeix a tots els àmbits: Restauració, amb un 16% del total, comerç/indústria, amb un 26% del total, i Llars, sent aquestes les líders amb un 58% del malbaratament.

Totes aquestes dades ajuden a entendre la magnitud d'un problema sovint imperceptible i ignorat a nivell individual però totalment inacceptable a nivell global en un món on 795 milions de persones, segons la WFP (World Food Programme), no tenen prou menjar per a mantenir un estil de vida saludable. Així doncs, per petita que sigui l'extensió de l'aplicació Appat i el percentatge de reducció que pugui suposar en el malbaratament de cada un dels seus usuaris, cada quilo comptarà per a reduir un malbaratament alimentari suposa una amenaça important per al medi ambient al nostre planeta.

12. Versions futures

Quan es desenvolupa una aplicació cada dia es tenen idees, més senzilles o més complexes, sobre coses que es poden millorar. Això fa que sigui molt difícil determinar un moment en el qual s'atura el procés de millora i es defineix una versió final però és necessari fer-ho per a poder fer proves i corregir els errors de l'aplicació. Per a l'aplicació Appat s'ha optat per definir el dilluns 6 de Juny com a última data per a la introducció de noves millores però deixant marge per acabar les que ja estaven començades. En aquest apartat es mostren totes aquelles millores que han quedat fora del termini dividides per categories.

12.1.Funcionalitats

L'aplicació Appat es basa en tres funcionalitats principals: La gestió dels menús, a través de les pantalles "Un sol àpat" i "Múltiples àpats"; la consulta de receptes, a través de la pantalla "Receptes" dins de "Crèdits, explicacions i receptes"; i l'esgotament d'un ingredient, a través de la pantalla "Em vull acabar...". Més enllà d'aquestes tres funcionalitats hi ha espai per algunes de noves sense abandonar la idea del projecte Appat. Algunes d'aquestes podrien ser les següents:

- **Secció de creació de salses.** Podria ser molt positiu per a l'aplicació un assistent per a la creació de salses ja que, especialment per a un sol àpat, un sol ingredient principal acompanyat per una bona salsa pot convertir-se en un gran àpat.
- **Pantalla de suggeriment d'espècies.** Tot i no tenir amb prou feines relació amb l'objectiu de combatre el malbaratament alimentari, un assistent per a l'elecció d'espècies podria ser especialment útil per donar matisos de gust als plats i, especialment, per permetre a l'usuari personalitzar aquests plats que a vegades no admeten variacions degut a la limitació d'ingredients.
- **Secció de postres.** Hauria d'estar separat de l'assistent normal de menús ja que, a diferència d'un primer o un segon plat, unes postres no pot ser un plat únic. No obstant, l'afegit de les postres als menús permetria donar major cobertura a l'assistent i, sobretot, esgotar més ingredients.
- **Secció d'edició de la base de dades.** Caldria crear una secció pròpia per a fer aquesta edició amb l'objectiu d'evitar que l'usuari faci canvis a la base de dades que després donin problemes en l'execució de l'aplicació.

Pel que fa a les millores d'aquestes pròpies seccions es podria afegir:

- **Possibilitat d'elecció d'un o dos plats.** A hores d'ara, Appat assumeix que els usuaris menjaran un únic plat per àpat. De totes maneres és positiu que els usuaris puguin escollir prendre dos plats perquè això ajudarà l'aplicació a arribar a més gent.
- **Possibilitat de determinar un temps màxim de cuina.** Es podria introduir tant a la secció "Múltiples àpats" com a "Un sol àpat" i la seva implementació seria molt senzilla. Només faria falta assignar un valor de temps per a cada plat i introduir-lo al principi o al final de la llista d'ingredients necessaris per a aquest plat. A continuació caldria afegir una comparació a les funcions de cerca entre aquest temps i el temps màxim donat per l'usuari. Per últim només faltaria modificar les funcions de cerca perquè ignoressin aquest primer o últim terme de les llistes.

12.2.Heurística

L'elecció d'una l'heurística determinada sempre es basa en trobar una solució de compromís entre velocitat de resolució i qualitat de resolució. No obstant, més enllà d'aquesta dicotomia hi ha marge real per a la millora a partir de la implementació de senzilles idees que poden agilitzar el procés en general. Alguns exemples en són:

- **Crear etiquetes generals per a conjunts d'ingredients.** Per exemple:
 - Verdures saltejables: Pebrot verd, pebrot vermell, carbassó, albergínia...
 - Verdures per bullir: Mongetes verdes, pèsols, coliflor, bròquil...
 - Fruites per a amanides: Pinya, maduixa, poma, magrana...
 - Embotits: Xorisso, llom, pernil salat...
 - Salsitxes: Frankfurt, Bratwurst, xistorra, cervela...
 - Pasta: Tallarines, spaghetti, macarrons...
- **Definició de semiacompanyaments.** Com s'ha definit a la secció 8, alguns plats necessiten d'un mínim d'ingredients però no importa quins per poder ser considerats plats. Per exemple una amanida de cigrons pot dur un munt d'ingredients com són cigrons, blat de moro, olives, ou, tomàquet o tonyina. És evident que els cigrons són indispensables, però pel que fa als altres cap és

imprescindible però tampoc es pot prescindir de tots. En aquests casos s'hauria de crear una nova distinció en els ingredients, com podria ser que s'escriuissin //nomIngredient//, i que el programa requerís de tenir-ne un mínim de 2 o 3 d'aquests per poder fer el plat. D'aquesta manera cap ingredient seria indispensable però no es retornaria que l'amanida de cigrons és possible només amb cigrons.

- **Diccionaris personalitzables.** La personalització dels diccionaris es faria en dues direccions, afegir coses al diccionari i treure-les o amagar-les.
 - **Diccionaris reduïbles.** L'usuari seria capaç de descartar plats i/o ingredients dels diccionaris de funcionament de l'aplicació. Això permetria a l'usuari descartar els plats/ingredients que no li agraden o els aliments als quals és al·lèrgic. Es podrien crear diccionaris reduïts estàndard per a celíacs, vegetarians, intolerants a la lactosa, vegans...
 - **Diccionaris oberts.** L'usuari seria capaç d'introduir ingredients a les receptes ja existents creant una secció específica per a això com s'ha comentat a l'apartat anterior, funcionalitats.
- **Gestió d'alternatives per a la cerca de múltiples àpats.** La primera versió d'Appat utilitza una metodologia molt senzilla per a presentar diferents alternatives. Bàsicament torna a executar la mateixa funció de cerca negant-li la possibilitat d'introduir el primer dels plats solució de la primera combinació. Aquesta metodologia, ni se n'hi pot dir heurística, és clarament millorable ja que aquesta pot produir fàcilment falsos negatius en cas que el primer dels plats fos indispensable per a la solució. En una següent versió es podria fer quelcom similar al que es fa a la funció "Un sol àpat", calculant i valorant diferents combinacions possibles en funció del nombre d'ingredients que s'esgoten.
- **Possibilitat de modificació dels plats per part de l'assignació d'acompanyaments.** Aquesta innovació podria quedar inclosa en la millora de l'anterior punt depenent de com es gestionés. La situació que resoldria és aquells casos en els quals el programa retorna un plat que es podria modificar a un de molt similar per a poder utilitzar més ingredients. Un clar exemple seria quan s'introdueixen ous i bacó com a ingredients, cas en el que el programa pot suggerir cuinar ou remenat i no utilitzar el bacó quan podria suggerir cuinar ous amb bacó i s'aprofitaria aquest ingredient.
- **Comptabilitzar l'impacte dels acompanyaments en les racions del plat.** El sistema Appat negligeix l'impacte dels acompanyaments en la quantitat de menjar

del plat. Aquesta negligència és correcta sempre i quan hi hagi pocs ingredients acompanyament, però a mida que se n'afegeixen més es torna més difícil ignorar el seu impacte. Així doncs seria interessant estudiar algun mètode per quantificar-ho com ara que a partir de 3 acompanyaments un plat es considera que “omple” un àpat i mig.

12.3. Interfície d'usuari

La interfície d'usuari és potser un dels punts febles de l'aplicació Appat i això es deu a dos fenòmens: D'una banda, no és objecte principal d'estudi del treball i, d'altra banda, s'han patit alguns problemes tècnics de funcionament del Kivy amb Windows. Aquests problemes tècnics han estat més abundants del que s'esperava i, tot i que molts s'han acabat resolent, han tingut cert impacte en l'aspecte final de la interfície.

El major dels problemes tècnics que no s'ha pogut resoldre és el posicionament lliure de widgets. No funcionen les propietats `pos_hint_x` i `pos_hint_y` i això impedeix fer distribucions de la pantalla que s'adaptin a la mida d'aquesta a no ser que s'utilitzi un `BoxLayout`, una tècnica senzilla però suficientment efectiva per a dur a terme el que es desitjava.

Més enllà dels problemes, les propostes de millora pel que fa a la interfície d'usuari són:

- **Creació d'una finestra a la cantonada superior dreta** que sigui present en totes les pantalles i que aportï les informacions de feedback que a hores d'ara l'usuari rep d'etiquetes posicionades diferentment en funció de la pantalla.
- **Implantació de text predictiu.** Quan l'usuari escriu els noms dels ingredients, que l'aplicació li suggereixi els noms dels plats a partir de les lletres que ha escrit.
- **Possibilitat de seleccionar les quantitats d'ingredients més comunes a través de botons.** Per evitar a l'usuari haver d'escriure la quantitat de la que disposa per a cada ingredient, es podrien col·locar botons per les quantitats més habituals per agilitzar el procés. De totes maneres caldria mantenir la possibilitat d'introducció a través del teclat pels casos en que fos necessari.
- **Possibilitat d'introduir els valors pressionant la tecla enter.** Es pot fer pesat per a l'usuari el fet d'haver de clicar el botó “següent ingredient” cada cop que vol introduir el següent ingredient.
- **Possibilitat de tornar enrere en la jerarquia de pantalles.** A hores d'ara hi ha

pantalles de l'aplicació on no es pot tornar enrere, sinó que o es va endavant o es torna directament al menú principal. Tot i que aquestes pantalles contenen la informació que l'usuari necessita, la navegació habitual en apps o pàgines web sempre permet a l'usuari tornar enrere i la manca d'aquesta possibilitat es podria percebre negativament per part de l'usuari.

12.4. Informació a l'usuari

És molt important que l'usuari de l'App tingui accés a tota la informació que desitja. Sovint quan s'està creant una App, es pensa en quines poden ser les necessitats de l'usuari i únicament se li mostra la informació que li soluciona aquestes necessitats. Tot i ser aquesta la manera de treballar que crea menys maldecaps al programador i a l'usuari, és especialment necessari en el món de l'alimentació que l'usuari pugui accedir a tota la informació d'una manera senzilla. Així doncs les propostes de millora d'aquesta secció aniran enfocades a donar a l'usuari certa informació.

- **Creació d'etiquetes per als plats.**
 - Apte per a vegetarians
 - Apte per a vegans
 - Apte per a intolerants a la lactosa
 - Apte per a celíacs
- **Informació de l'aportació calòrica d'una certa quantitat del plat.** Recordar que aquest punt és complicat ja que Appat treballa amb racions i no amb quantitats, cosa que pot fer variar les proporcions dels ingredients significativament i tenir un gran impacte en l'aportació calòrica

12.5. Altres

Queda fora de les altres seccions estipulades la utilització de l'aplicació Bulldozer per a poder utilitzar Appat en un dispositiu mòbil com s'ha comentat a la secció 7.5.2.

13. Conclusions

13.1. Compliment d'objectius

Els objectius inicials es poden consultar a la secció 3.1.

Pel que fa al compliment d'objectius de l'alumne, aquests s'han complert ja que s'ha aconseguit desenvolupar una aplicació amb un funcionament satisfactori i capaç de desenvolupar les tasques que s'esperaven d'ella. No obstant, si es va a l'objectiu ètic, que és la reducció del malbaratament alimentari, encara no s'ha complert ja que depèn de què l'aplicació s'utilitzi i això dependrà en principal mesura de la implementació d'aquesta en telèfons mòbils. És cert que l'aplicació d'ordinador es podria difondre igualment però, a hores d'ara, la seva utilització requereix de la instal·lació de Python i de la llibreria Kivy, procés que no és senzill i no està a l'abast de tot el públic.

Pel que fa als objectius del tutor, l'alumne no és la persona més adient per parlar-ne però, des d'un punt de vista objectiu, s'ha aconseguit aplicar prou satisfactòriament la dobleta Python + Kivy en un sistema operatiu Windows tot i haver hagut de resoldre múltiples problemàtiques.

13.2. Aprenentatge

És sabut que el desenvolupament de tot projecte comporta un cert aprenentatge, però el fet és que desenvolupar l'aplicació Appat ha estat especialment enriquidor per a l'alumne. Això es deu a la quantitat de diferents àmbits en els que s'ha treballat ja que, a diferència de projectes molt concrets com podrien ser la optimització d'un procés industrial, la creació d'una aplicació implica treballar en molts camps diferents com poden ser software, logística, organització, disseny... Es creu que l'experiència que s'ha adquirit en aquests camps pot ser de gran valor en el futur laboral de l'alumne. Possiblement la prova més tangible d'aquest aprenentatge és el convenciment de què si es tornés a començar el projecte es farien les coses diferent i s'obtindrien resultats similars en molt menys temps. A tall d'exemples, l'alumne voldria comentar algunes d'aquestes lliçons que ha après.

- **Dedicar-se simultàniament a aprendre Kivy i a desenvolupar l'app és la millor manera d'aprendre però la pitjor per desenvolupar l'aplicació.**

És sabut que la millor manera d'aprendre és posant els propis coneixements en pràctica, cosa que ha passat en el moment de desenvolupar el projecte. El problema és que quan

es desenvolupa un projecte d'aquestes característiques és molt important conèixer exactament de quines eines es disposa per a dur-lo a terme i això no succeeix fins que l'usuari és un expert en aquestes eines. Un clar exemple pot ser que en un primer moment es va resoldre el problema de la mida del text de les etiquetes copiant la comanda "text_size : (nomEtiqueta.width, None)" a cada una de les etiquetes per acabar descobrint unes setmanes més tard que era possible crear una classe personalitzada que heretés de la classe "Label", definir allà aquesta propietat i crear etiquetes personalitzades directament.

- **És molt important saber filtrar les fonts de coneixement.**

El filtratge de les fonts de coneixement i l'avaluació d'aquestes és quelcom que un alumne ha de fer al llarg de tota la carrera i es pot dir que aquesta habilitat condicionarà en gran mesura els seus resultats acadèmics o, com a mínim, la seva proporció resultats/dedicació. Aprendre el llenguatge Kivy no és una excepció ja que a l'hora d'aprendre'l es disposava de la pàgina web i el llibre creating apps in Kivy, però la lectura d'aquestes fonts no va resultar ser gaire enriquidora per a l'alumne. No va ser fins que va descobrir els vídeo-tutorials de l'usuari de youtube Sentdex, que l'alumne va començar a assolir una comprensió suficient sobre el llenguatge Kivy.

- **La planificació és clau**

"Una hora avui te'n pot estalviar deu demà". Aquesta frase sintetitzaria el punt de vista de l'autor sobre tot el que és el desenvolupament d'una app o de qualsevol programa en general. Sovint es té la tendència natural de treballar sobre la marxa i, tal i com s'indica en la primera lliçó apresada, utilitzar l'experiència que es va adquirint per a millorar el procés. Doncs bé, que aquesta tendència pot generar enormes pèrdues de temps quan s'està treballant amb programació ja que en aquest entorn és indispensable tenir una bona organització, encara que sigui esquemàtica, del codi i el que s'espera d'aquest. Malauradament, com ja s'ha dit, aquesta premissa és incompatible amb l'aprenentatge sobre la marxa i en el projecte Appat s'han patit en pròpia pell els problemes que resulten d'intentar canviar coses quan el projecte ja està massa avançat i les funcions estan tant interrelacionades entre elles que es complica detectar on s'originen els errors de funcionament. Un clar exemple és la introducció dels plats-acompanyaments. Quan es van voler definir els plats-acompanyaments es va decidir que s'escriurien de la següent manera a la base de dades: `**nomPlatAcompanyament**` com a plats i `^nomPlatAcompanyament^` com a ingredients. En un primer moment s'esperava que adaptant la funció `nomIngredient` per als plats escrits amb aquest format seria suficient per a que l'aplicació funcionés sense cap problema. Malauradament va acabar sent necessari localitzar errors i canviar detalls en mínim tres de les funcions principals, cosa que va provocar un munt de feina.

Encara que ho sembli, aquest concepte no és contradictori amb el desenvolupament en forma de metodologia àgil que se li ha donat a l'aplicació ja que cada una de les iteracions de la metodologia àgil pot tenir la seva planificació i, fins i tot pot haver-hi una planificació general que es vagi comprovant a cada iteració.

- **Relativitzar els encerts i errors quan es treballa en codi**

Durant el desenvolupament del TFG, Appat ha donat grans alegries però també grans disgustos i un d'aquests moments extrems es va produir al desenvolupar les proves. A mida que s'havien anat creant les funcions també s'havien anat testejant satisfactòriament, de manera que s'entenia que poques coses podien anar malament i que les proves de funcionament serien gairebé un tràmit. Res més lluny de la realitat doncs la primera volta de les proves, com es pot veure a la secció 9.3, va acabar amb un 5/11 de funcions plenament satisfactòries per la secció "Un sol àpat" i un 14/24 per a la secció "Múltiples àpats". Aquests resultats no eren els que s'esperaven i van resultar una dura topada amb la realitat. Afortunadament, quan es va començar a treballar en aquests errors un per un es va veure que alguns tenien el mateix origen i que, no sense dedicar-hi una bona quantitat de temps, tots eren possibles de resoldre. D'aquí en queda com a lliçó el relativitzar els encerts, quan les funcions creades es testeaven satisfactòriament, i relativitzar també els errors, quan les proves van resultar nefastes, doncs una senzilla modificació a una funció pot alterar totalment el funcionament d'aquesta.

- **Definir versió final i no afegir-hi res més**

Quan es desenvolupa una app des d'un punt de vista no professional sempre hi haurà funcions millorables o detalls a afegir. Això no seria un problema si no fos perquè aquesta perspectiva de millora contínua no casa amb el concepte de data límit. En el cas del projecte Appat s'ha hagut de fixar una data a partir de la qual no s'afegien funcionalitats a l'app, per senzilles que fossin, ja que sinó no era possible fer les proves de funcionament i acabar de redactar la memòria. A més cal dir que sovint quan s'introdueixen millores cal fer petites variacions d'algunes de les funcions existents per evitar fallades col·laterals, de manera que no sempre és una tasca ràpida.

13.3.Observacions finals

L'aplicació Appat per si sola no serà prou efectiva en la lluita contra el malbaratament alimentari si no va acompanyada d'una compra responsable. L'aplicació només pot resoldre situacions puntuals d'aliments que estan a punt de caducar però si aquesta

situació es torna endèmica, possiblement sigui conseqüència d'un problema de compra excessiva que l'aplicació no pot resoldre. També és necessari que l'usuari tingui una bona gestió de les sobres alimentàries de casa, ja que la introducció lliure d'acompanyaments pot acabar generant uns plats excessivament grans i, si les sobres d'aquests acaben a les escombraries, s'està aconseguint el contrari del que es desitjava.

Per cloure la memòria es pot dir que, l'aplicació Appat pot ser molt útil en la lluita contra el malbaratament alimentari però que per a ser realment efectiva requereix d'un usuari compromès en aquesta lluita i amb uns bons hàbits de compra. Així doncs, com sempre, quedarà a les mans de les persones el millorar la situació del medi ambient al món i, per moltes eines que se'ls aportin com podria ser Appat, l'únic remei infal·lible és i serà l'educació en els valors.

Agraïments

Agrair a Lluís Solano Albajes, tutor del projecte, la col·laboració i la paciència durant aquests mesos.

Agrair a la meva padrina i cosina Ariadna Puig Gabarró la seva col·laboració, doncs fou ella qui va suggerir el nom d'Appat.

Bibliografia

Referències bibliogràfiques

- [1] PHILLIPS.D, *Creating Apps in Kivy*. O'Reilly, ISBN: 978-1-491-94667-1
- [2] SMARTDRAW, *SmartDraw, draw flowcharts & diagrams online*, webapp per al dibuix de diagrames de flux.

[<https://www.smartdraw.com/> , 09/06]
- [3] PYTHON, *The official home of the Python Programming Language*, pàgina web del llenguatge de programació Python.

[<https://www.python.org/> , 10/06]
- [4] KIVY, *Kivy: Cross-platform Python Framework for NUI Department*, pàgina web del mòdul Kivy.

[<https://kivy.org/#home> , 10/06]
- [5] STACK OVERFLOW, *Stack Overflow, A language-independent collaboratively edited question and answer site for programmers*, pàgina web de preguntes i respostes, independents del llenguatge de programació, per a programadors editada col·laborativament.

[<http://stackoverflow.com/> , 10/06]
- [6] WHATIS, *WhatIs.com*, pàgina web de definicions tecnològiques.

[<http://whatis.techtarget.com/> , 09/06]
- [7] SUPERCOOK, *Supercook.com*, webapp d'organització de menús i receptes.

[<http://www.supercook.com/#/recipes> , 08/06]
- [8] EPICURIOUS, *Epicurious.com*, pàgina web de cuina, menús i receptes.

[<http://www.epicurious.com/> , 08/06]
- [9] ALLRECIPES, *Allrecipes.com*, webapp d'organització de menús i receptes.

[<http://allrecipes.com/> , 08/06]

- [10] AN. J, *Hack your way to dinner with these apps that make sense of your fridge*, article de divulgació sobre aplicacions i webs relacionades amb la gestió de menús.

[<http://www.digitaltrends.com/web/what-can-i-make-tonight-apps-to-make-deciding-what-to-cook-easier/#:zDoTE2dSg3t03A> , 06/06]

- [11] MINISTERIO DE INDUSTRIA, *Protección de la propiedad intelectual*, documents informatius sobre la protecció de la propietat intel·lectual.

[http://www.oepm.es/es/invenciones/herramientas/manual_del_inventor/proteger_su_idea/ , 09/06]

- [12] SPEEDPATENT, *Speedpatent.es*, pàgina web d'una companyia de creació de patents.

[<http://www.speedpatent.es/?gclid=CNKFnvXR8MwCFQso0wod5DIMgw> , 10/06]

- [13] TECMARK, *Tecmark patentes y marcas*, pàgina web d'una companyia de gestió de la propietat intel·lectual.

[<http://tecmark.es/ca/> , 10/06]

- [14] LA FÁBRICA DE INVENTOS, *La Fábrica de Inventos: empresa que desarrolla, patenta y comercializa tus ideas*, pàgina web d'una companyia de patents i comercialització.

[<http://lafabricadeinventos.com/servicios/redaccion-y-registro/?gclid=CMuL3YHS8MwCFYZuGwodil4HsQ> , 10/06]

- [15] SUGRAÑES, *Sugrañes: Agents i advocats de la propietat industrial i intel·lectual*, pàgina web d'una companyia de gestió de la propietat intel·lectual.

[<http://www.sugranes.com/> , 10/06]

- [16] HEDA, *Heda patentes y marcas*, pàgina web pàgina web d'una companyia de gestió de la propietat intel·lectual.

[<http://es.hedapatentes.com/contacto-paw/?gclid=CNrVx47S8MwCFUQcGwodDmkFDg> , 10/06]

- [17] WEBOPEDIA, *How much will a server cost?*, article sobre costs de servidors.

[http://www.webopedia.com/DidYouKnow/Hardware_Software/how_much_will_a_server_cost.html , 10/06]

- [18] PYMES Y AUTÓNOMOS, *El coste del servidor en la pyme: conclusiones*, article sobre costs de servidors.

[<http://www.pymesyautonomos.com/tecnologia/el-coste-del-servidor-en-la-pyme-conclusiones>, 10/06]

- [19] EMPRENDEDORES.ES, *¿Cuánto cuesta lanzar una app?*, article sobre els costos de llençar una app al mercat

[<http://www.emprendedores.es/gestion/crear-app/crear-app-3>, 10/06]

- [20] ÀREA METROPOLITANA DE BARCELONA, *Malbaratament alimentari*, pàgina informativa sobre el malbaratament alimentari

[<http://www.amb.cat/web/medi-ambient/residus/prevencio/reduccio/malbaratament-alimentari> , 12/06]

- [21] FOOD AND AGRICULTURE ORGANIZATION OF THE UNITED NATIONS, *SAVE FOOD: Global Initiative on Food Loss and Waste Reduction*, pàgina web amb dades estadístiques sobre el malbaratament alimentari.

[<http://www.fao.org/save-food/resources/keyfindings/en/> , 12/06]

- [22] PAYSCALE, *PayScale – Salary Comparison, Salary Survey, Search Wages*, pàgina web amb dades estadístiques sobre salaris.

[<http://www.payscale.com/>, 10/06]